ISSUE NUMBER 15 $1.50 in NORTH AMERICA ($2.50 elsewhere) NOVEMBER 1976

BYTE

the small systems journal

INTED IN USA



It's More Fun Than Crayons

SWTPC 6800-SOFTWARE

WARNING — It has been determined that reading this ad may be hazardous to your health, if you own another type computer system. We will not be responsible for ulcers, heartburn, or other complications if you persist in reading this material.

4 K BASIC© — 8 K BASIC©

* Full floating point math
* 1.0E-99 to 9.99999999E+99 number range
* User programs may be saved and loaded
* Direct mode provided for most statements
* Will run most programs in 8K bytes of memory (4K Version) or 12K bytes of memory (8K Version)
* USER function provided to call machine language programs
* String variables and trig functions—8K BASIC only

COMMANDS	STATEMENTS			FUNCTIONS		
LIST	REM	END		ABS	† VAL	† SIN
RUN	DIM	GOTO*	STOP	INT	† EXT$	† COS
NEW	DATA	ON. . .GOTO*	GOSUB*	RND	† LEN$	† TAN
SAVE	READ	ON. . .GOSUB*	PATCH*	SGN	† LEFT$	† EXP
LOAD	RESTORE	IF. . .THEN*	RETURN	CHR	† MID$	† LOG
PATCH	LET*	INPUT	† DES	USER	† RIGHT$	† SQR
	FOR	PRINT*	† PEEK	TAB		
		NEXT	† POKE			

* *Direct mode statements*
† *8K Version only*

MATH OPERATORS	
-	(unary) Negate
*	Multiplication
/	Division
+	Addition
–	Subtraction
† ↑	Exponent

RELATIONAL OPERATORS	
=	Equal
< >	Not Equal
<	Less Than
>	Greater Than
< =	Less Than or Equal
>=	Greater Than or Equal

Your BANKAMERICARD welcome here

WE HONOR master charge THE INTERBANK CARD



You guys are out of your minds, but who am I to complain. Send —

☐ 4K BASIC CASSETTE $4.95
☐ 8K BASIC CASSETTE $9.95
☐ MP-68 Computer Kit $395.00

NAME

ADDRESS

CITY STATE ZIP

Southwest Technical Products Corp.
Box 32040, San Antonio, Texas 78284

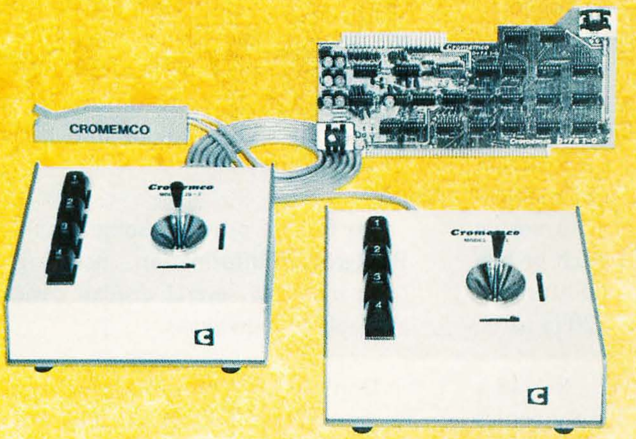


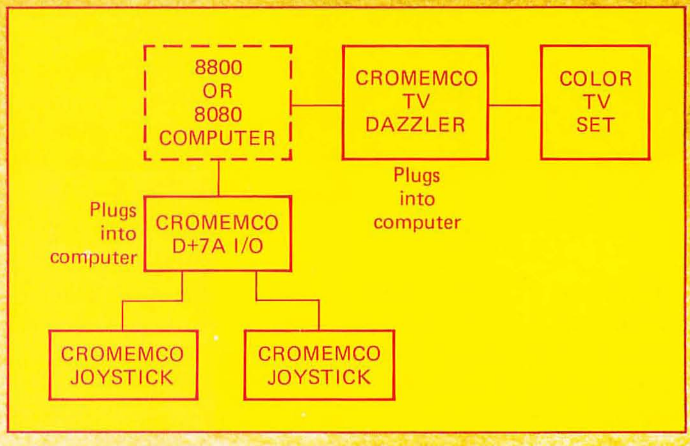







You get more games, more fun, more computer uses with this new joystick





AND THERE'S AN EASY WAY TO INPUT IT TO YOUR COMPUTER

You'll get a lot more fun out of your computer with this new joystick.

But note that it is not just an ordinary joystick — it is a *console*. It has a 2-axis joystick *and* contains a *speaker* and *speaker amplifier*. You can have *sound* with your games or, say, warning sounds in other applications. Or have your computer talk to you.

Gives you sound, too

A third feature you get is *four pushbutton switches*. These give you even more possible uses such as selecting various colors on a color graphics terminal.

Four pushbuttons

1
2
3
4

EASY TO COUPLE

To couple the new joystick to your computer, just use our D+7A™ I/O board. It will couple not only *one* but *two* consoles to your Altair™ 8800 or IMSAI 8080. And you'll still have several analog channels left over (and one 8-bit output port).

The D+7A plugs into the Standard 100 (S-100) bus of your Altair or IMSAI computer.

EASY TO DISPLAY

Displaying the joystick outputs with the software below is also easy. Just use our TV DAZZLER™ board. It also plugs into the S-100 bus.

NEW SOFTWARE

Here's some new Cromemco software for the joystick (to display, use DAZZLER interface):

CHASE! (2 persons, 2 joysticks): the cross chases the circle. The vertical bars move downward to add more fun to the chase. Score and remaining time kept automatically.

TRACK (1 person, 1 joystick): move the dot to the center of the spiral without touching the spiral's arms.

DAZZLE DOODLE (1 person, 1 joystick): lets you draw pictures in 4 colors on your color TV terminal using the joystick.

STORE/MAIL

Cromemco wishes you more fun, more use from your computer. Get this new joystick console and other Cromemco peripherals at your computer store or order from the factory.

Joystick console kit (Model JS-1K)	$ 65
Joystick console assembled (Model JS-1W)	$ 95
D+7A I/O kit (Model D+7A-K)	$145
D+7A I/O assembled (Model D+7A-W)	$245
TV DAZZLER kit (Model CGI-K)	$215
TV DAZZLER assembled (Model CGI-W)	$350

SOFTWARE
(Punched paper tape with documentation)

CHASE!	$15
TRACK	$15
DAZZLE DOODLE	$15

PROMPT SHIPMENT

Mastercharge and BankAmericard accepted with signed order. Please show complete card number and expiration date.

Mail orders shipped prepaid if fully paid with order. Purchase orders accepted subject to credit approval.

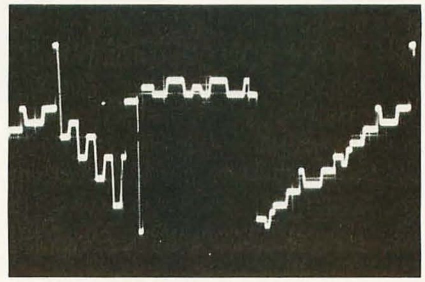
California users add 6% sales tax.

Cromemco

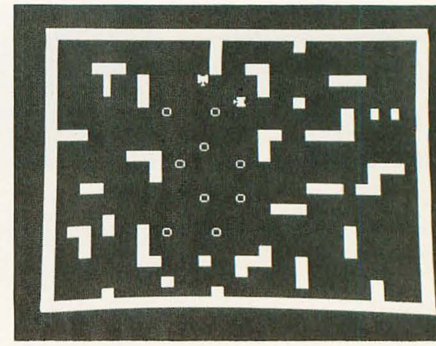
Specialists in computers and peripherals

2432 CHARLESTON RD., MOUNTAIN VIEW, CA 94043 • (415) 964-7400

In This



page 78



page 32



page 26

It's More Fun Than Crayons! Read how Mike and Alex Rosner, ages 7 and 5, discovered the joys of computer art and incidentally proved the need for timesharing computers in the home. On the cover are super stars Mike and Alex at work.

If Isaac Newton were alive today, he'd be immersed in long and complicated physics manipulations, which are ultimately tested using calculations involving multidimensional matrices and matrix operations of linear algebra. Chances are he might be interacting with a computer, with an interpreter from the APL tree of languages. Read Mark Arnold's **What Is APL?** to find out a bit about APL, a language with a mathematical orientation but utility for general purpose programming as well.

How do you do graphics in a weekend, without spending a fortune? In **Beer Budget Graphics**, Peter Nelson tells all: how the synergistic combination of a couple of digital to analog converters and an output latch or two brings the world of point plotting to an oscilloscope for about $20 in parts.

In the July 1976 BYTE, we ran a functional specification for a graphics interface in answer to reader Paul Hyde Jr's letter. Practically before the ink had dried on the July press run (figure of speech) reader Thomas R Buschbach sent in this article on how he made just such a graphics interface for his Digital Group system using existing timing logic of the television display unit. Given an existing 8 K memory, processor and television display generator, this high resolution display can be added to a system for as little as $25 in additional semiconductors.

Star Trek freaks should turn to Joe Deres' article to find **An Enterprising Display**. In the article you'll find information on a design for a television graphics output which can be built from a kit (or your own parts) for less than $100.

Ira Rampil provides **Some Graphics Background Information,** including a comparison of several display devices available to individuals.

Don't be satisfied with simple visual readouts. **Make Your Next Peripheral a Real Eye Opener** by implementing a true vector display using an XY oscilloscope, circuits and software similar to Steve Ciarcia's design. In his article you'll see how to draw a picture of a moderately high resolution Star Ship Enterprise.

Build This Video Display Terminal, advises Alfred Anderson, who took C W Gantt Jr's TV interface circuit described in the June 1976 BYTE, made a few changes, added a few goodies, and wrote a few lines of software. Now he tells you how you can do it.

page 106



What's NOT In This BYTE

. . . Software Bug of the Month, Book Reviews, Ask BYTE, a slew of What's New and BYTE's Bits items we were dying to print, lots of good letters, and some dynamite articles. Even using smaller type in places didn't enable us to include much that we urgently wished to. If you missed it this month, watch for it next.

In the Queue

Foreground

Background

Nucleus

BYTE magazine is published monthly by BYTE Publications Inc, 70 Main St, Peterborough New Hampshire 03458. Subscription rates are $12 for one year, $22 for two years, $30 for three years in North America. Elsewhere: $15 one year, $25 two years, $33 three years. Second class postage paid at Peterborough NH 03458 and at additional mailing offices. Phone 603—924-7217.

Address editorial correspondence to Editor, BYTE, 70 Main St, Peterborough NH 03458.

BYTE #15

NOVEMBER 1976

staff

Imagine a microcomputer

Imagine a microcomputer with all the design savvy, ruggedness, and sophistication of the best minicomputers.

Imagine a microcomputer supported by dozens of interface, memory, and processor option boards. One that can be interfaced to an indefinite number of peripheral devices including dual floppy discs, CRT's, line printers, cassette recorders, video displays, paper tape readers, teleprinters, plotters, and custom devices.

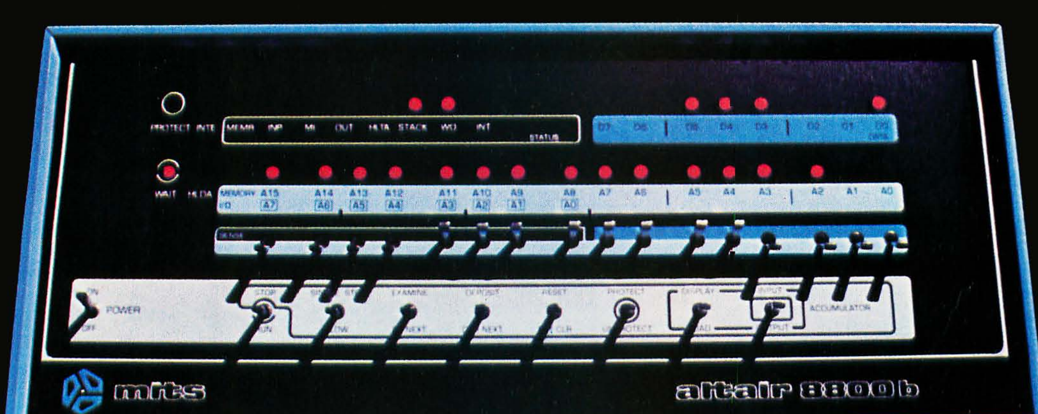
Imagine a microcomputer supported by extensive software including Extended BASIC, Disk BASIC, DOS and a complete library of business, developmental, and industrial programs.

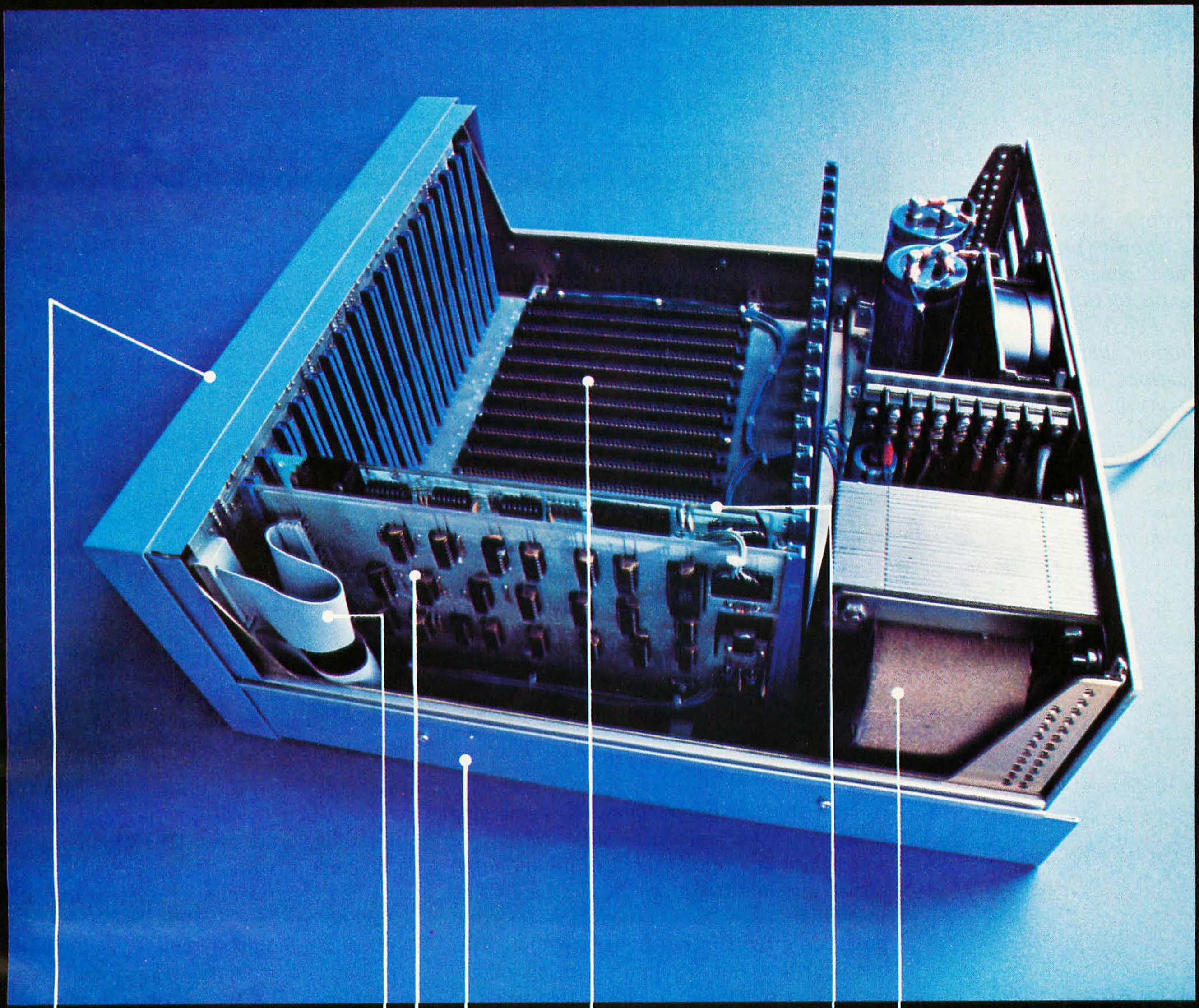
Imagine a microcomputer that will do everything a mini will do, only at a fraction of the cost.

You are imagining the Altair™ 8800b. The Altair 8800b is here today, and it may very well be the mainframe of the 70's.

The Altair 8800b is a second generation design of the most popular microcomputer in the field, the Altair 8800. Built around the 8080A microprocessor, the Altair 8800b is an open ended machine that is compatible with all Altair 8800 hardware and software. It can be configured to match most any system need.

NOTE: Altair is a trademark of MITS, Inc.





Redesigned front panel. Totally synchronous logic design. Same switch and LED arrangement as original Altair 8800. New back-lit Duralith (laminated plastic and mylar, bonded to aluminum) dress panel with multi-color graphics. New longer, flat toggle switches. Five new functions stored on front panel PROM including: DISPLAY ACCUMULATOR (displays contents of accumulator), LOAD ACCUMULATOR (loads contents of the 8 data switches (A7-AO) into accumulator), OUTPUT ACCUMULATOR (Outputs contents of accumulator to I/O device addressed by the upper 8 address switches), INPUT ACCUMULATOR (inputs to the accumulator from the I/O device), and SLOW (causes program execution at a rate of about 5 cycles per second—for program debugging).

Full 18 slot motherboard.

Rugged, commercial grade Optima cabinet.

New front panel interface board buffers all lines to and from 8800b bus.

Two, 34 conductor ribbon cable assemblies. Connects front panel board to front panel interface board. Eliminates need for complicated front panel/bus wiring.

New, heavy duty power supply +8 volts at 18 amps, +18 volts at 2 amps, 18 volts at 2 amps. 110 volt or 220 volt operation (50/60 Hz). Primary tapped for either high or low line operation.

New CPU board with 8080A microprocessor and Intel 8224 clock generator and 8216 bus drivers. Clock pulse widths and phasing as well as frequency are crystal controlled. Compatible with all current Altair 8800 software and hardware.

altair 8800-b

MITS, Inc. 1976/2450 Alamo S.E./Albuquerque, New Mexico 87106



Photo 1: Alex Rosner, age 5, shown preparing the graph paper coding sheet for his picture from a pencil sketch. Once the correlation between picture locations and commands was learned, both children stopped planning layouts in advance on the coding sheets. The "snow with house and mailbox" is shown in photo 5.

It's More Fun

Richard Rosner
Pocono Rd
Brookfield CT 06804

Now that the home computer is emerging from its crystal radio stage and now that we are becoming more applications oriented rather than hardware oriented, we are going to find that we have competition for computer time right in our homes. While I have been putting microcomputers using microprocessors together for almost two years, it wasn't until I hooked up the TV to my present system that my family took interest. Now, when I walk in the door after work, the kids corner me asking which one can use the computer first.

The microcomputer system uses a MOS Technology 6502 CPU, a TIM (Teletype Interface Monitor), and 1 kilobyte of programmable memory. The graphics module contains a 2 K by 8 programmable memory allowing a 128 by 128 point display. In these pictures the display is configured

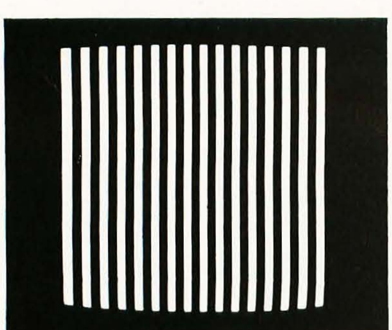


Photo 3: "Cage," by Alex Rosner, freehand.



Photo 4: Rocket Ship With Moon, by Mike Rosner.

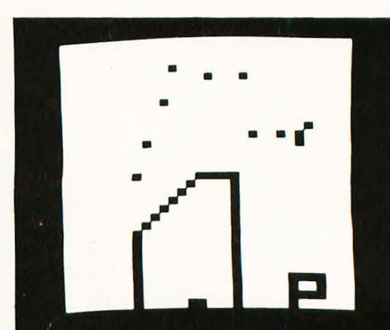


Photo 5: Snow With House and Mailbox, by Alex Rosner.



Photo 2: Michael Rosner, age 7, sitting at the Teletype entering data for the image of a rocket ship with moon (see also photo 4).

Than Crayons

(under software control) as a 32 by 32 array.

The data for the display is entered serially. The software commands are

R, reset to top line,
N, go to beginning of next line,
W, enter a white square,
K, enter a black square, and
H, halt (return to monitor).

The procedure is to take a piece of graph paper with a 32 by 32 grid blocked off and to then draw a picture by filling in some of the squares. After the drawing is complete the paper tape with the program is loaded into the processor and the data is entered with the above five commands using the drawing as a coding sheet.

This, of course, is not an ideal hardware or software approach for a video interface. The interface was built for another application and the processor was used as a test

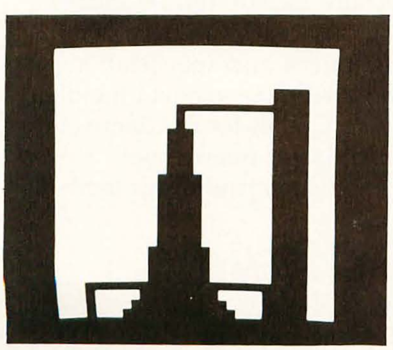


Photo 6: Rocket Ship, by Mike Rosner.

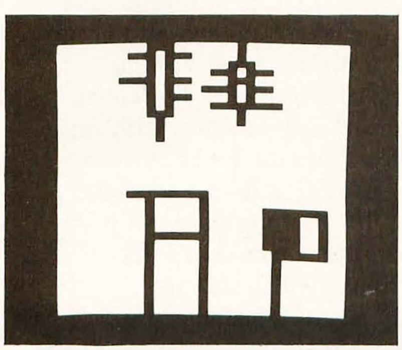


Photo 7: Fun Play (Swing Set, Man, Sun and Mail Box), by Alex Rosner.

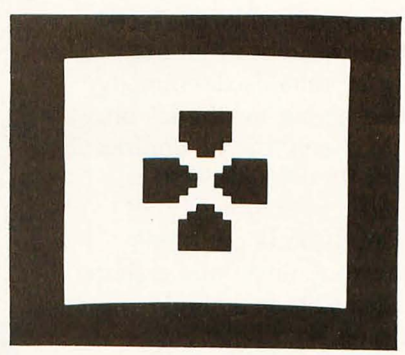
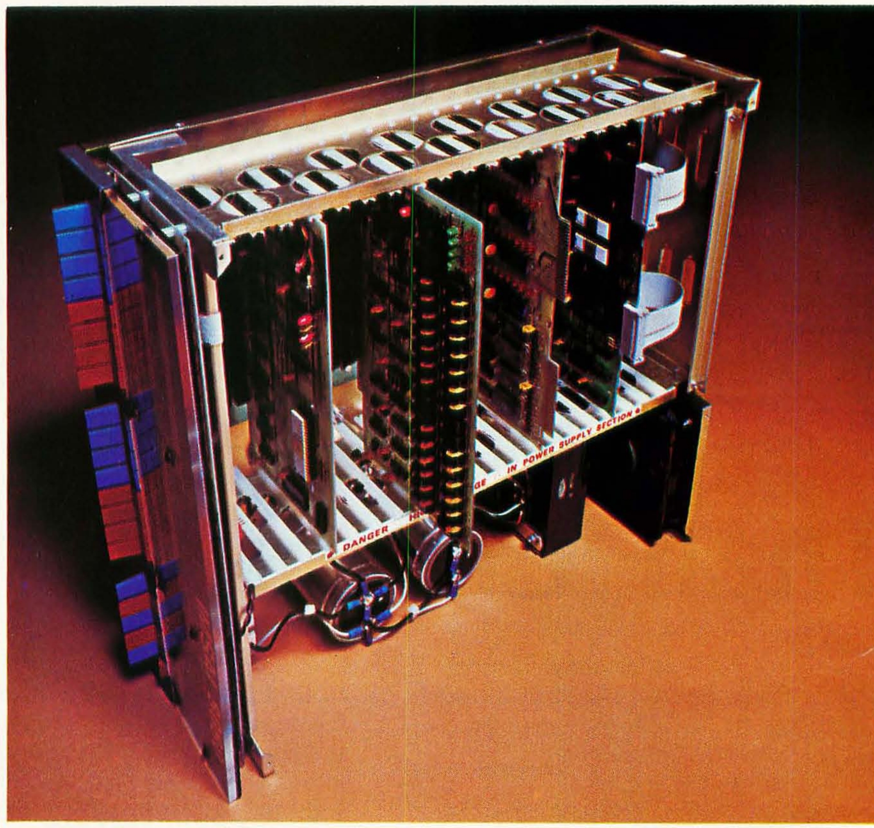


Photo 8: Flower Puzzle, by Mike Rosner, freehand.

If you thought a rugged, professional yet affordable computer didn't exist,



think IMSAI 8080.

Sure there are other commercial, high-quality computers that can perform like the 8080. But their prices are 5 times as high. There is a rugged, reliable, industrial computer, with high commercial-type performance. The IMSAI 8080. Fully assembled, it's $931. Unassembled, it's $599. And ours is available now.

In our case, you can tell a computer by its cabinet. The IMSAI 8080 is made for commercial users. And it looks it. Inside and out! The cabinet is attractive, heavy-gauge aluminum. The heavy-duty lucite front panel has an extra 8 program controlled LED's. It plugs directly into the Mother Board without a wire harness. And rugged commercial grade paddle switches that are backed up by reliable debouncing circuits. But higher aesthetics on the outside is only the beginning. The guts of the IMSAI 8080 is where its true beauty lies.

The 8080 is optionally expandable to a substantial system with 22 card slots in a single printed circuit board. And the durable card cage is made of commercial-grade anodized aluminum.

The IMSAI 8080 power supply produces a true 28 amp current, enough to power a full system.

You can expand to a powerful system with 64K of memory, plus a floppy disk controller, with its own on-board 8080–and a DOS. A floppy disk drive, an audio tape cassette input device, a printer, plus a video terminal and a teleprinter. These peripherals will function with an 8-level priority interrupt system. IMSAI BASIC software is available in 4K, that you can get in PROM. And a new $139 4K RAM board with software memory protect. For the ultimate in flexibility, you can design the system for low-cost multiprocessor, shared memory capability.



Find out more about the computer you thought didn't exist. Get a complete illustrated brochure describing the IMSAI 8080, options, peripherals, software, prices and specifications. Send one dollar to cover handling.

Call us for the name of the IMSAI dealer nearest you.

Dealer inquiries invited.

IMSAI

IMS Associates, Inc. **Dept. B-11**
14860 Wicks Boulevard
San Leandro, CA 94577
(415) 483-2093

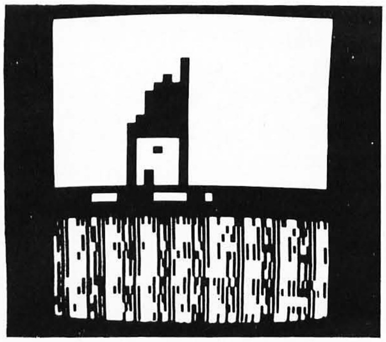


Photo 9: House, by Alex Rosner.

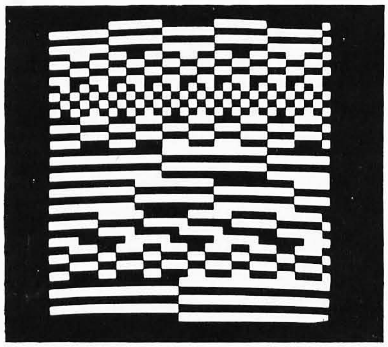


Photo 10: Illusion Number 4, by Mike Rosner, freehand.

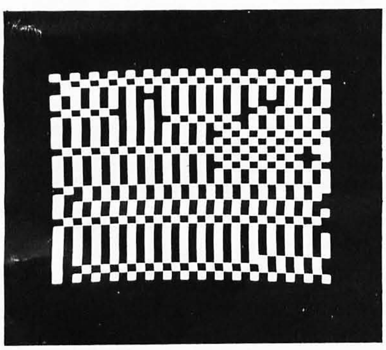


Photo 11: Tree Record, by Alex Rosner, freehand.

fixture. After testing the display unit, I put a little "HELLO MOM" and a starship on the screen and showed it to my wife and two kids. The children immediately wanted to do the same thing and we now have a new toy.

The older child, Michael, is seven and learned to generate a picture and use the program after about two hours of instruction. He is shown in photo 2 as he puts his third creation on the screen, a rocket with the moon. Above his right hand can be seen the drawing he is working from. (The pattern in the lower portion of the TV screen is the random data in the programmable memory chips when the power is turned on.) All his efforts at keypunching have been saved on paper tape for delayed replay when Grandma comes to visit.

The younger child, Alex, five, is shown in photo 1 as he prepares the coding sheet from a sketch for his third picture – a house with mailbox and snow. Alex required a considerable amount of help with his first two pictures, but he quickly learned to use the graph paper correctly.

The cover photo shows the entire setup. Alex is putting a picture on the TV while Mike is using BRUIN (a language similar to BASIC). It seems that second graders are now being taught the rudiments of algebra. When my calculator batteries went dead I called up my company's computer to do some simple calculations. Mike saw me using the CRT terminal and wanted to try it. While we can't call this programming, he is having fun.

The collection of photos from the TV screen shows some of the work the kids have been doing along with the titles. The inception, generation and titling of all the pictures are the kids' own without help from adults. Mike's first three pictures were made using work sheets as were Alex's first three. The rest were made without first making drawings.

The microcomputer is set up in the family playroom. I consider myself quite fortunate in that I have never had to fight with my wife for space. From the beginning of this venture I have had my wife's full support. Perhaps this is because (as she has recently informed me) she plans on using the computer to help her in the various business courses in which she is enrolling, as well as for household functions such as diet menu planning.

If you are as lucky as I am, you may find yourself putting up a signup sheet as the family competes for computer time.■



Photo 12: Mike's first work, no title.

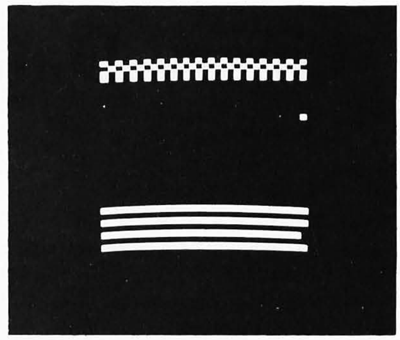


Photo 13: "Black Cat" by Alex Rosner, freehand.

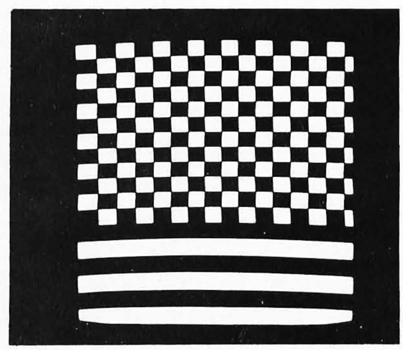


Photo 14: "Illusion Number 1" by Mike Rosner, freehand.

Editor's Note: *Readers will detect a careful balance between the amount of photographic exposure given to Alex and Mike. Author Richard Rosner reports that the realpolitik of family life requires equal treatment for both children in his article. . .* **CH**

Walter Banks and Roger Sanderson of the University of Waterloo, Waterloo, Ontario CANADA, have proposed a method of publishing machine readable programs in magazines such as BYTE. This proposal was the subect of some discussion in a technical session at the Personal Computing 76 show in Atlantic City. I wrote the present report, based upon preliminary written information provided by Walter and Roger, and telephone discussions with Walter on August 15 1976. The technical content of this report, with the exception of the proposal of format 3 and addition of a frame identification field to the block format, is supplied by Walter and Roger. . . . CH

A Proposed Standard for Publishing Binary Data in Machine Readable Form

by
Walter Banks
Computer Communications Network Group
Roger Sanderson
Dept of Electrical Engineering
University of Waterloo
Waterloo, Ontario CANADA

and

Carl Helmers
Editor, BYTE magazine

Purpose

The purpose of this technology is to provide a method of encoding binary data in the form of printed marks on a magazine, book or other published page. These printed marks are to be in a predefined format which can be read by a simple and inexpensive optical scanning head which drives a single bit input port of a personal computer; decoding will be done by means of software in the computer. The goal is to provide a means of disseminating program information in a machine readable form.

Physical Formats

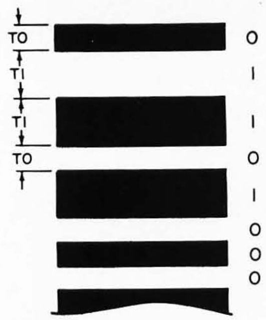
Figure 1 details three possible physical formats. These are proposals at the present time. Other formats are possible and should not be excluded from contention; readers are invited to make proposals. The formats are described in the captions of figure 1. The following comments can be made.

- Format 1, bar width modulation with alternate light and dark, is the most compact method of encoding. Its density varies with the detailed content, but for a given minimum light (or dark) interval measurable by the input routine and a given binary string of data, this method will be higher density than either format 2 or format 3 below. A disadvantage is that emergency manual reading of the data with a magnifier is probably more difficult due to the need to interpret four states: wide space, wide mark, narrow space, narrow mark. This method had been proposed for use in the retail trade by Research Inc, Box 24064, Minneapolis MN 55424, some time ago, before the UPC (Universal Product Code) scheme was adopted for grocery product checkout use.
- Format 2, ratio recording, has an advantage in certain points of view in its fixed width. Being fixed width, the amount of data per page is fixed, as is the amount of data per line of machine readable type. However, it is a lower density form than either format 1 or format 3 below. It can probably be read by eye with greater ease than format 1 since every bit is present as a short or long bar which is actually printed, and only two states need be interpreted.
- Format 3, fixed gap bar width modulation, is basically a higher density version of ratio recording. Take the layout of format 2, and compress the gaps between black marks to a fixed length, and you will get format 3. Since this format is higher density, it may be preferable to format 2, while retaining the advantages of having a human readable black mark for every bit.

Frame Layout and Page Layout

Figure 2 shows the layout of a single frame of information, interpreted within the

Figure 1: Three possible formats. These are proposals at this time, with the appropriate parameters indicated.



FORMAT I

Figure 1a: Format 1: Bar width modulation, with alternate dark and light. The parameters to be specified in printing are widths T0 and T1. A trailing bit complementary to the last data bit in a string is required. The leading bit of a string will be assumed to be in the dark state. In reading this code, time between transitions falls into two categories, long for a 1 bit, and short for a 0 bit.

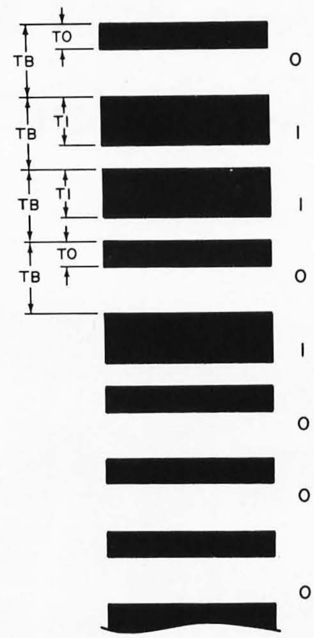


Figure 1b: Format 2: Ratio recording. The parameters to be specified in printing are the bit length, TB and the T0 and T1 widths for states of the data. This format has a fixed length per bit which is independent of the state of the data. In reading this code, the time from one light to dark transition to the next light to dark transition is the duration of the bit cell, which is compared to the duration of the dark period to find out whether a 1 bit (long) or 0 bit (short) was read.

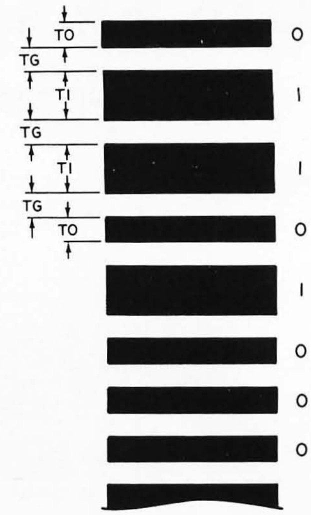


Figure 1c: Format 3: Fixed gap bar width modulation. The parameters to be specified in printing are the interbar gap width TG, and the bit length parameters T0 and T1. In reading this code, the length of the interbit gap gives a calibration for judging the next dark period as a 0 bit (short) or 1 bit (long). The data density (as in format 1) varies with the statistics of the number of 0s and 1s in a given region of the printed data.

string as a synchronization pattern followed by 8 bit bytes of information.

The SYNCH pattern is an 8 bit character which must occur as the first byte of every frame, and may be preceded by several timing bits to initialize adaptive software. Once the input string equals the SYNCH character, the decoding algorithm can proceed bit by bit with the rest of the frame.

The CHECKSUM field contains an 8 bit check sum of the data in the current frame. This is the primary error detection technique employed. In manual scanning, when an error occurs, the sensor head can be drawn

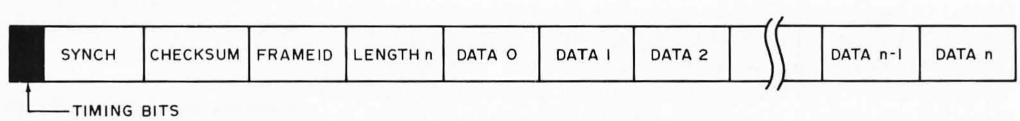


Figure 2: Frame Format. The timing bits may not be required, but provide a preliminary leader before the ASCII SYN character (hexadecimal 16) which leads off the frame. All data is transmitted most significant bit first. Following the synchronization character is an 8 bit check sum representing 2's complement modulo 256 summation of all the remaining data on the record. The FRAMEID field is an 8 bit integer used for relative positioning within an extended file. Its purpose is to allow manual rescan in the event of errors, so that the software will recognize the input as the same record. The length field contains a direct integer value for the number of bytes in the data field. From 1 to 255 bytes can be in the data field; a length of 0 is reserved for a special case "end of file" frame. Finally, the remainder of the frame contains 8 bit bytes of data.

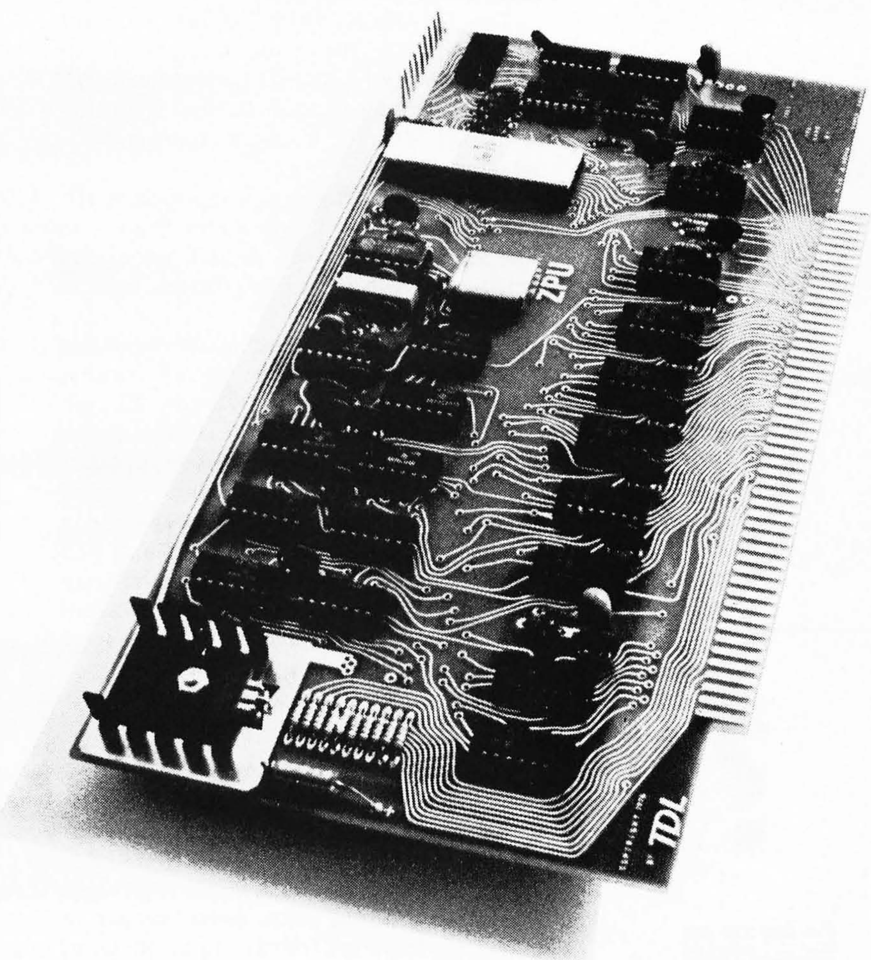
We've got it! Z-80 power for the Altair bus.

Here it is, TDL's ZPU™ the highest point of technology for an Altair/IMSAI system. Now, you can multiply your present capabilities without creating costly obsolescence. Take advantage of the wide range of existing hardware backup for your current system. The ZPU is compatible and dependable with many plus features you'll want to know about.

To further round out your system we also have available the fastest and lowest power static ram boards going (from 4 to 16K with expansion) and a system monitor board with a Z-80 monitor, powerful I/O and audio casette features.

As for software, TDL's user support program is unmatched in the micro-processor industry, currently available are the Z-80 monitor, basic, and the most sophisticated MACRO-ASSEMBLER yet developed.

Join the Z-80 revolution — it's more than just hardware!



ZPU Kit $269.00
Z16K Kit $574.00

Dealer Inquiries Invited.

HOW TO ORDER Just send check or money order, or use your BankAmericard or Mastercharge, and your orders will be shipped to you postpaid. COD orders must be accompanied by a 25% deposit. Your credit card order must include the serial # of the card, expiration date, and your order must be signed. New Jersey residents add 5% state sales tax. For more information, send for our free catalog.

TDL

(609) 392-7070

TECHNICAL DESIGN LABS, INC.
342 COLUMBUS AVENUE
TRENTON, NEW JERSEY 08629

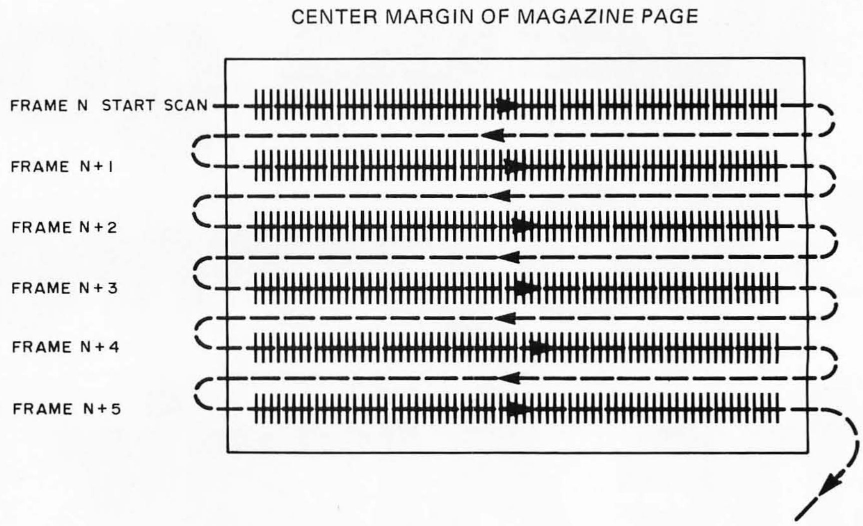


Figure 3: Page Layout. A page of data in printed form [as we would run in BYTE] would be laid out with the data running parallel to the center margin of the magazine. This allows the maximum amount of data in a frame, thus minimizing the overhead bits of the frame format.

past the data again without moving the straightedge used as a guide.

The FRAMEID field contains an 8 bit identification of the current frame within an extended block of data. This identification is required in order to allow the user an option to reload data if a check sum error is detected.

The length of the data is specified by the LENGTH field as a byte count from 1 to 255 (actual counts will depend upon data density, but will most likely be less than 50 per frame).

Figure 3 shows the page layout to be used in a typical magazine page size of 7 inches wide by 10 inches high (18 cm by 25 cm). From his experimental work, Walter Banks reports that a bit density of 50 bits per inch (20 bits per cm) in the ratio recording method looks quite reasonable. Based on this linear density, and putting 1/5 inch (0.5 cm) spacing between frames in the horizontal direction, the page capacity of the layout in figure 3 can be calculated:

Bits per frame = 50 BPI * 10" = 500
Bits per page = 5 frames per inch * 7" * 500 bits per frame = 17,500

Overhead of the frame format in figure 2, excluding timing bits, is 32 bits per frame, so the total data capacity is 17,500 – 35 * 32 = 16,380 bits. In bytes, this is 16,380/8 = 2047.5 bytes. The actual frame parameters would be adjusted to an even number of bytes, whatever the format employed. Using the fixed format, each page can hold 2 K (2048) bytes with the density figures reported. Shrinking the interbit gaps to the format 3 fixed size would probably give an improvement of 5 to 15 percent depending upon the width parameters chosen and the statistics of the data being presented in this form.

How Is the Data Generated?

The secret of this whole method at the magazine production end is using modern phototypesetting equipment to create the images of the data which are used in the magazine master layouts. When (for example) an article describing an advanced high level language interpreter written by a university or basement lab hacker is to be printed, the paper tape or magnetic tape text of the program and its relocatable loader would be converted into phototypesetting commands for the typesetting equipment, on 9 track tape or paper tape compatible with the typesetter. Once the copy is received from the typesetter, the rest of the layout is standard operating procedure for any magazine – keep track of the order of the program pages and add any auxiliary copy needed to identify pages or segments within the program. With 2 K bytes per page, dissemination of an 8 K to 10 K package would only require four or five pages in the magazine, and would be easily accomplished. We expect to perform experiments with this sort of dissemination technique in future editions of BYTE.

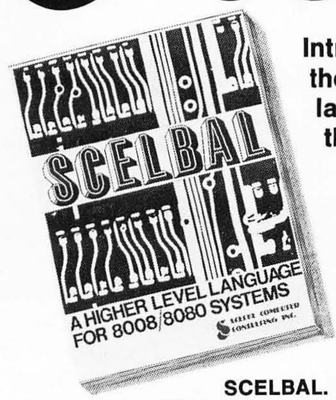
Will It Work?

The experiment looks quite reasonable, but there are numerous possible hitches which can be considered. For example, will the data density projected (50 bits per inch, 20 bits per centimeter) be practical? What will be the incidence of errors due to ink splotches and other glitches? Will the simple conception of a reading head (photo transistor, TTL level converter and straight edge as a guide) provide the signal needed? This report is written on the basis of preliminary experiments; further information on this subject will appear in future BYTEs.■



A low density test string.

SCELBI SOFTWARE



Introducing SCELBAL, the new microcomputer language that's simpler than machine language.

A complete language for "8008"/"8080" systems including source listings, routines, flow charts and more!

SCELBAL. SCientific **EL**ementary **BA**sic **L**anguage for "8008"/"8080" systems. A complete, illustrated program book. Routines. Techniques. Source Listings. Flow Charts. And more. Took several years to develop. Now yours for many years to come. First time that intimate details of higher level language has been offered for such a low price. Only $49! You get **5 Commands:** SCR, LIST, RUN, SAVE, LOAD. **14 Statements:** REM, LET, IF . . . THEN, GOTO, FOR with STEP, END, INPUT, PRINT, NEXT, GOSUB, RETURN and optional DIM. **7 Functions:** INT, SGN, ABS, SQR, RND, CHR, TAB. And, it runs in 8K and more. Here's all the data needed to customize a high level language for your system . . . at a fraction of the cost!

Order your copy today! Get started advancing your system! $49 ppd.

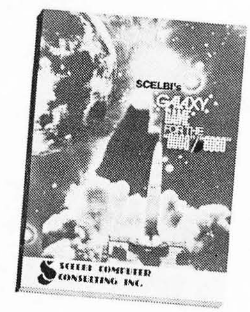


Here's SCELBI's First Book of Computer Games for the 8008/8080. Action-packed. And fun.

Try to beat the computer at its own game. Here's the first complete machine language computer manual for computer games to include source listings, flow charts, routines and more. **Space Capture** — You against the computer using "search and destroy" strategy to shoot down roaming alien spaceships in outerspace. **Hexpawn** — a mini-chess game that lets the computer make mistakes . . . but only once. **Hangman** — an updated version of the great kid game. Computer selects words at random from long, expandable list. Try to beat it in 8 moves or less. Illustrated. Fun extras to put your computer to challenging, competitive, fun use. **Order yours today! 14.95 ppd.**

SCELBI's new microcomputer Game Book of Outer Space . . . GALAXY

Captain your own crusading starship against the logic of your "8008" or "8080". You must destroy a random number of alien ships without running out of stardates, out of fuel, out of ammunition or out of the galaxy. Plan your mission. How much fuel is used for your warp factor? Don't run into roaming stars that can damage your ship. **Suddenly! Condition Red! Alien in sight!** How big is he? Fire a phasor or torpedo! He's damaged or destroyed. But, you've used valuable fuel. That's just the beginning. A new game every time. Complete program book in machine language for 4K memory, including source listings, flow charts, routines, etc. Great intergalactic adventure and fun.



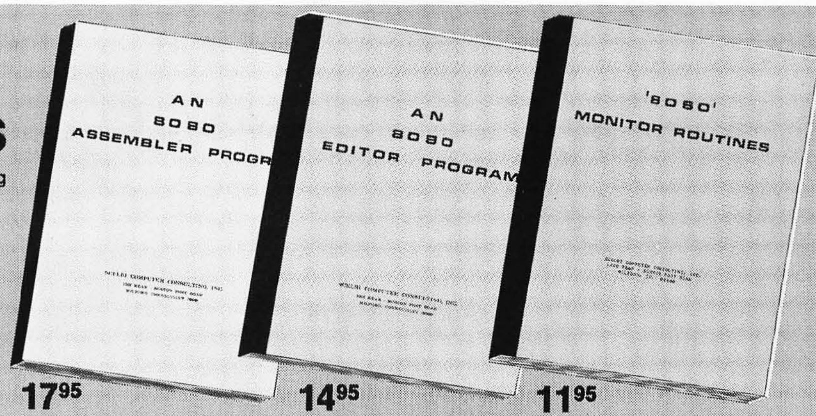
Order "GALAXY" today! 14.95 ppd.

"8080" SOFTWARE MANUALS

Three new basic, complete "MUST" manuals! **"8080" Assembler Program** operates easily in 4K bytes of RAM (including symbol table). Unique feature: How to accept mnemonics related to "8008" based machines on "8080". Includes all major routines, pertinent flow charts, highly commented assembled listing and more. **$17.95.**

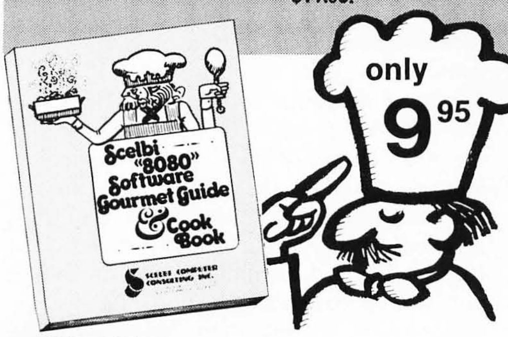
"8080" Editor Program identical to "8008" Editor, with machine codes for "8080". **$14.95.**

"8080" Monitor Routines same functions as "8008" specifically developed to utilize expanded capabilities of "8080". **$11.95.**



17.95 ppd. 14.95 ppd. 11.95 ppd.

SAVE $5.35! BUY ALL 3 FOR ONLY $39.50!



Now you can cook-up hot programs for your own "8080"

Gobble up practical "how to" facts, including the description of "8080" instruction set. How to manipulate the "8080" stack. Flow charts and source listings. General purpose routines for multiple precision operation. Programming time delays for real time applications. Random number generators. And many other basic programming function comestibles.

The "Cook Book" even includes a completely assembled floating point arithmetic program . . . plus input/output processing for your basic I/O programming through interrupt processing. Code and numeric conversion routines. Real time programming. Search and sort routines. And many, many more finger-lickin' goodies. Order your copy of **Scelbi's "8080" Software Gourmet Guide & Cook Book** today. Only $9.95 ppd.

Bon appetite!

Prices shown for North American customers. Master Charge, Postal and Bank Money Orders preferred. Personal checks delay shipping up to 4 weeks. Pricing, specifications, availability subject to change without notice.

SCELBI COMPUTER CONSULTING INC.

1322 Rear Boston Post Rd., Milford, CT 06460
Telephone: 203/874-1573

by Carl Helmers

The Address Space Saturation Problem

(or, What Happens After You Use 32 16 K Dynamic Memory Chips?)

According to an article by Marvin Gold in the September 6 1976 edition of *Electronic News*, the 16 K dynamic memory chips are almost ready for volume production, and price estimates are quoted. According to Mr Gold's article, the 1976 average selling price of 16 K memory chips will be an estimated $16 each, and when volume production levels are reached in 1977, the average price at the factory will be $12. Assuming a personal computing company which has moderate volume production purchases the 16 K memories at $12, its component cost for a 16 K memory segment is $96, and for a 32 K memory segment is $192 in mid to late 1977. Multiply this by a factor of 3 to 5 to take into account packaging, manufacturer profit margins, and merchant markup; and it can be expected that in mid to late 1977 the personal computer with 32 K memory will have a memory subsystem cost of perhaps $600 to $1000 at the retail store.

The designers of the early microprocessors never conceived of the large scale integration computer as anything more than a controller in dedicated applications (so the grapevine says). The rumor which was running around some time ago was that one LSI firm's reaction to the idea of a FORTRAN (or other high level language) for their microcomputer was "who'd ever want to do such a silly thing." The history of small general purpose computers implemented with LSI processors since then demonstrates that a lot of people want to do exactly that, and more: APL, SNOBOL, LISP, PASCAL, ALGOL, TRAC™ etc, etc. Similarly, the reaction currently might be "who'd ever want to put more than 64 K bytes of memory in a personal computing system?" My claim is that there are plenty of people who would want to (in order to

Continued on page 65

Personal Computing 76:

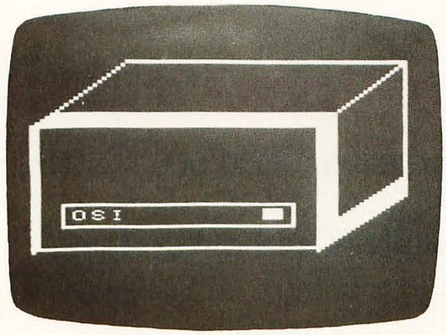
Outstanding Computer Hobbyist of the Year Award

by
John Dilks
Cochairman of PC 76

At the Personal Computing 76 show, the members of the committee organizing the show felt it would be most appropriate to create an award for a person who has made some outstanding contributions to the progress of computing by individuals. We selected the person, then dreamed up the award. The person is Sol Libes, president of the Amateur Computer Group of New Jersey. The award consists of a plaque for Sol to keep, and a second plaque with spaces for future entries to be engraved. This is how traditions are started.

We had numerous reasons for wanting to present this award to Sol. Sol is employed as an educator, and is a true amateur interested in using the computer. He is one of the principal organizers of the Amateur Computer Group of New Jersey, seeing it grow from a near handful of people last year to over 400 members with typically 200 attending the many meetings. He is a leader in computer education, where at the Union County Technical Institute he has created a hands-on training program in microcomputer technology and has set up courses which could easily serve as prototypes for similar courses throughout the country. He set up and ran the Trenton Computer Festival of the ACGNJ, one of the first such happenings ever. And, during the past summer, he and his family travelled throughout the USA, meeting with amateur computer clubs and club leaders and manufacturers both to find out what's happening and to further the development of the field. We think these reasons more than justify the recognition, which was a complete surprise to Sol and his family at the banquet August 28. We hope to continue this award and tradition at future Personal Computing shows.■

Meet the Challenger.™



The Challenger
Self Portrait

The new price and performance champ from OSI.

He's got his act together!

Even our lowest-cost Challenger comes fully assembled, complete with a 500 ns 6502A, serial interface, 1,024 words of memory and a UL-approved power supply, all for $439. Every Challenger comes ready for easy expansion with an 8-slot mother board, backplane expansion capability, and a power supply heavy enough to handle a full complement of system boards. Our 4K Challenger comes ready to run BASIC minutes after you unpack it. And there's more.

He packs some heavy hardware.

You've never seen memory and interface options like these—not at our prices, fully assembled! 4K RAM memory boards $139! (see below). Single drive OSI Challenger Floppy Disk $990! Dual drive Floppy $1490! Plus 8K PROM boards! A Video Graphics board, including alphabetics, graphics, and **color!** An audio cassette, A/D, D/A and parallel I/O board! A backplane extender board! A prototyping board! And our extraordinary **CPU Expander Board**—it lets you run a Z-80, and 6100 (PDP-8 equivalent) **concurrently** with The Challenger's 6502, or under its control.

There's nothing soft about his software!

OSI has full software support for our Challengers. Including extended BASIC, extended Video Monitor, a Disk Operating System, some very Hollywood real time programs for Video Graphics, Animation, Sound Processing and so forth, plus PROM firmware, with more to come.

He's fast!

You can order The Challenger with a 6502C for a 250 ns cycle time, with a standard 6502A for 500 ns cycle time, or with a 6800 for 1 microsecond cycle time. And with our CPU Expander Board, you can always update to any new CPU to be as fast as fast can be.

And he isn't just good!

He's better! By design. The OSI Challenger is the only completely-assembled, ultra-high-performance, fully-expandable mainframe computer that does this much for this little. Get your hands on one now. Send for your Challenger today.

You can't beat The Challenger!

The OSI Challenger 65-1K. Fully assembled. Features 6502A CPU, serial interface, 1,024 words of memory. **$439.**

The OSI Challenger 65-4K. Same as 65-1K but with 4,096 words of memory. Will run Tiny BASIC without expansion. **$529.**

The OSI Challenger 65V-4K. NO NEED for an expensive terminal. Connects to your ASCII keyboard and video monitor through included OSI 440 Video Board. Features software utility that simulates a deluxe CRT terminal. **$675.**

The OSI Challenger 68-1K. Based on 6800 CPU. For the casual hobbyist, smaller systems. The Challenger 68 series comes only in serial interface forms and is compatible with MIKBug software through an included OSI software utilities package. **$459.**

The OSI Challenger 68-4K. With OSI 4K BASIC on paper tape. **$529**

SPECIAL! ADDITIONAL 4K MEMORY BOARDS. Ordered with your Challenger, limit 3 more at this special Low Price, (total 16K, including 4K already on-board in mainframe). **$139**

Buy 12K or larger Challenger 65 system and we include Extended BASIC FREE!

OSI Challenger Floppy Disk System. Fully assembled, for use with OSI Computers only. **$990** Single drive **$1490** Dual drive.

OSI Audio Cassette Interface. Comes assembled, but with room for you to populate with A/D and D/A chips later. (OSI 430 based) **$89**

And all the baseboards and kits of the powerful OSI 400 System.

OK, OSI, I'm ready to buy!

To order your Challenger System, send the total amount of your purchase plus $4.00 for shipping and insurance (plus sales tax for Ohio orders) by personal money order or check. Or indicate **all** numbers on your BankAmericard or Master Charge to charge your order.

Or send a 20% (non-refundable) deposit to receive your order C.O.D.

Delivery is typically 60 days (except when payment is by check, which must clear before shipment can be made). Deliveries are scheduled on a first ordered, first shipped basis.

Name____________________

Address____________________

City__________ State__________ Zip__________

Telephone____________________

Bank card info Inter Bank #__________

Expiration Date____________________

Account #____________________

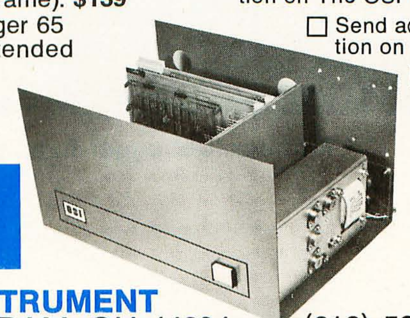
Check ☐ M. O. ☐ BAC ☐ MC ☐
20%, bal. C.O.D. ☐

☐ Order attached.

☐ Send additional information on The OSI Challenger.

☐ Send additional information on OSI 400 Kits.

☐ $1.00 enclosed for complete OSI Catalog.



OSI

OHIO SCIENTIFIC INSTRUMENT

Dept. B 11679 HAYDEN STREET, HIRAM, OH 44234 (216) 569-7945

HAVE A MERLIN

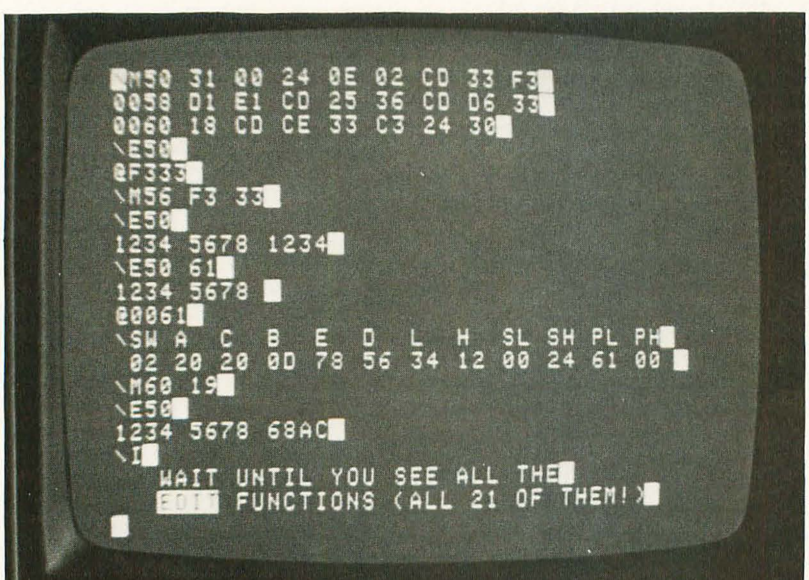
The Intelligent Video Interface

If your favorite computerist has an 8080 bus system, what better way to say Merry Christmas than with a MERLIN Video Interface? MERLIN is not just a superb character display . . . MERLIN is a whole system . . .

MERLIN is a text display,
a high density graphics display,
a keyboard interface,
a serial I/O port,
a regulated power source,
a RAM/ROM memory and
an integrated and expandable software package.

MERLIN is a gift that keeps giving. MERLIN's keyboard and I/O connector allows the MERLIN owner to add a variety of MiniTerm hardware expansion products. MERLIN's software has been written as an expandable ROM system with many general purpose subroutines available for use. Give your favorite computerist (yourself, perhaps?) a MERLIN Christmas!

Order early to insure delivery for Christmas.



Monitor/Debugger Example

FIRMWARE SPECS

MERLIN'S BASIC INTELLIGENCE (MBI) ROM has

Monitor Functions

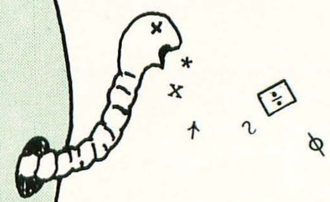
* Memory Fill with HEX value
* HEX Memory Dump
* ASCII Text Input
* HEX Memory Input or Modify
* Set Display Format
* Examine/Modify CPU registers
* Copy Memory Blocks
* Define Display Memory Area
* Four User Defined Functions
* Execute User Program
* Three Auto-Clear Breakpoints

Editing Functions

* Winking Keyboard Cursor
* Cursor up, down, rt, lt, & Home
* Insert and Replace Mode
* Delete Character and Line
* Delete to end of 'Display Memory'
* Four Slave Cursor Functions
* Auto *or* Manual Scroll
* Home/Clear
* Six User Defined Functions
* Page Up and Page Down

Readers Beware!

APL is often thought of as heavy mathematical stuff. Terminology used in this article reflects this background and is summarized in the glossary; table 1 should be referenced when reading the text to identify operations coded by the special symbols of APL. The symbols used in this article are sometimes approximations of a true "APL" ball for a Selectric typewriter. Many of the symbols are printed on the typical APL terminal using multiple overstrikes (ie: the sequence: character 1, backspace, character 2 . . .).



What Is APL?

Mark Arnold
1400 Grand Av
Laramie WY 82070

APL is the modest name of A Programming Language, created by Kenneth Iverson. APL is simple to learn, yet in many ways it is the most powerful language in use today. While to the untrained eye APL looks like the strangest computer language around, with the possible exception of LISP, the structure of APL is fairly close to that of more common languages such as FORTRAN or BASIC. This article will look at some of the features of a large system APL. Also, I will explore what features I think a small APL system for a microcomputer could have.

Writing in FORTRAN is like writing how to go downtown: You don't have to think a lot about what parts of the language you use; you just tell exactly what to do. Writing in APL is like writing poetry: You want to use the exact part of the language that will express what you want without any excess. APL haters criticize APL because there are so many operators for the user to remember, and the code doesn't always flow out without some thought, but APL enthusiasts like APL because the large number of operators greatly reduces the size of programs.

APL is extremely time-efficient (at least on large computers) for parallel operations, such as adding two sets of numbers together. Suppose your program was to figure out the bowling average of each of the four players on your team. In most languages you would have to use a loop which would go around for each time there was a new score. In APL, all you would need (provided the program was set up right) would be a +.

APL operates as an interpreter; that is, it looks at the APL language program and does the operation (perhaps after some intermediate translation). This is different from a compiler which translates to machine language before the program is run. Because of this, there are two states in which you operate APL. The first is called evaluated input and the second is called function definition. "Function" is the APL name for what would be called a "program" in other languages. A workspace is all the functions and variables that you have created. There can be more than one function in a workspace, and often you can save the workspace on an auxiliary storage device.

The evaluated input mode is as simple as 1+1. When evaluated input is in control, APL prints six spaces and requests input. If you type 1+1, APL will print out the sum of 1 and 1. It would look like this (where the indented part is what you type and the answer is printed out starting at the left hand margin)

```
      1+1                USER →
2                        ← APL
```

Notice that no "Print Statement" is neces-

About the Author:

Mark Arnold is a student at the University of Wyoming, Laramie WY. His experiences to date include co-authoring the SCELBAL interpreter program with Nat Wadsworth. He is currently at work on the definition and implementation of an APL subset interpreter for microprocessor systems.

sary as it would be in BASIC. Some common operators are +, −, x, ÷. These are the normal math symbols for the basic operations, but not the normal BASIC and FORTRAN symbols. The original APL used these and many more strange characters for its operators, and since the first APL systems were implemented on IBM/360s, this was no major problem. Changing a character set simply meant changing balls on a Selectric terminal. Most of the operators have two meanings (as summarized in table 1 for all APL operators). The example above of adding 1 to 1 is known as dyadic; that is, the addition operator has two operands: the one on the left and the one on the right. The − is an example of a monadic operator that has only one operand on the right:

```
      -3                          USER ⟶
¯3                                ⟵ APL
```

The −3 calculates the minus of 3. APL responds with ¯3, which is its way of saying "negative three." Note: The ¯ is used only with numbers, while the − can be used in any expression.

APL scans an expression from the right to the left which is opposite the way most high level languages do. This means 2−3−4 yields 3 instead of ¯5. *[In APL, a raised hyphen indicates an algebraic negative sign and an ordinary hyphen is used for subtraction.]* Also unlike most high level languages, there is no algebraic hierarchy in APL. So 2x3+4x5 is 56 and not 26. However, as in other high level languages, parentheses can be used to override the right to left convention: (2x3) +4x5 is 26.

All the expressions shown up to now have been of scalar values, that is, single numbers. APL also allows vectors as constants. A vector is an ordered set of scalar values, something like an array in FORTRAN or BASIC. So to figure the average of the two scores of the four members of your bowling team:

```
      .5 x 160 109 300 240        USER ⟶
      + 203 157 254 165           USER ⟶
181.5 133 277 202.5               ⟵ APL
```

where player one bowled 160 and 203, player two 109 and 157, and so on. The first thing APL does is to add the two vectors (scores) in parallel to get 363 266 554 405. Then each one of the four sums is multiplied by .5 to produce: 181.5 133 277 202.5, the average of those two games. All of this was done without an explicit loop or any program in the evaluated input mode. (Note: To do an operation, either the size of the two

Table 1: APL Operators. This is a complete listing, abstracted from the sources in the bibliography of this article, of the APL operators implemented in a typical large scale system such as an IBM/360 or IBM/370. The IBM 5100 machine is reported to run the full set of APL functions by emulating a 360/370 architecture in microcode and using close to 200 K bytes of interpreter ROM. A practical small system APL would necessarily be a subset if the interpreter is to be kept to a small size.

Monadic Scalar Operators

+	Identity: Returns the right operand as though the operator had not been there.
−	Minus: Returns the negative value of the right operand. (Appears above normal position of minus sign as in ¯3.)
x	Signum: Returns 1 if the right operand is greater than zero, 0 if it is equal to zero, and ¯1 if it is less than zero.
÷	Reciprocal: Returns the reciprocal of the right operand.
⋆	Exponential: Returns e raised to the right operand.
⍟	Natural Logarithm: Returns the LOG base e of the right operand.
⌊	Floor: Returns the greatest integer less than or equal to the right operand.
⌈	Ceiling: Returns the least integer greater than or equal to the right operand.
|	Absolute value: Returns the absolute value of the right operand.
!	Factorial: Returns the factorial of the right operand.
○	Pi times: Returns Pi times the right operand.
∼	Not: Returns a 1 if the right operand is 0, 0 if the right operand is 1; an error condition results if the operand is not logical (1 or 0).
?	Roll: Returns a random number between 1 and the operand.

Monadic Mixed Operators

,	Ravel: Converts a higher dimensioned array into a vector.
⍳	Index generator: Returns a vector of consecutive integers between 1 and the right operand.
ρ	Shape: Returns the length of a vector or the dimensions of a higher dimensioned array.
⍋	Grade up: Returns a vector that when used to index the right operand will sort it into ascending order.
⍒	Grade down: Returns a vector that when used to index the right operand will sort it into descending order.
∊	Execute: Returns the value of an expression represented by character data as though the characters value of the right operand had been entered during evaluated input.
⌹	Matrix inverse: Returns the inverse of the matrix specified in the right operand.
⍉	Monadic transpose: Returns a matrix such that the rows of the right operand are the columns of the new matrix.
⌽	Reversal: Returns a vector (if the right operand is a vector) where the elements of the vector are in the reverse order of the right operand.

Dyadic Scalar Operands

+	Addition: Returns the sum of the right and left operands.
−	Subtraction: Returns the difference of the right and left operands.
x	Multiplication: Returns the product of the right and left operands.

operands must be the same or one must be a scalar and the other an array. An operation such as + or x is called a scalar operator because it obeys this rule.)

Table 1, continued:

÷ Division: Returns the quotient of the right and left operands.

* Exponentiate: Returns the left operand raised to the right operand.

⍟ Logarithm: Returns the logarithm whose base is the left operand, of the right operand.

⌊ Minimum: Returns the minimum of the two operands.

⌈ Maximum: Returns the maximum of the two operands.

| Residue: Returns the remainder resulting from the division of the left operand by the right operand.

! Combinations: Returns the binomial coefficient of the left and right operands.

○ Circular functions: Returns the function of the right operand where the left operand specifies which function:

Left operand	Function of X	Function of −X
0	$\sqrt{1-X^2}$	same
1	Sin X	Arcsin X
2	Cos X	Arccos X
3	Tan X	Arctan X
4	$\sqrt{1+X^2}$	$\sqrt{X^2-1}$
5	Sinh X	Arcsinh X
6	Cosh X	Arccosh X
7	Tanh X	Arctanh X

< Less than: Returns 1 if the left operand is less than the right operand, and 0 otherwise.

≤ Less than or equal to: Returns 1 if the left operand is less than or equal to the right operand, 0 otherwise.

> Greater than: Returns 1 if the left operand is greater than the right operand, 0 otherwise.

≥ Greater than or equal to: Returns 1 if the left operand is greater than or equal to the right operand, 0 otherwise.

= Equal to: Returns 1 if the left and right operands are equal, 0 otherwise.

≠ Not equal to: Returns 1 if the right and left operands are not equal, 0 if they are.

∧ And: Returns 1 if both left and right operands are 1, 0 if either or both are 0. If one of the operands is not 1 or 0 then an error results.

∨ Or: Returns 1 if either one or both of the operands is 1, 0 if both are 0. If operands are not logical (1 or 0) an error will result.

⍲ Nand: Returns 0 if both right and left operands are 1, 1 if either one or both are 0. If operands are not logical an error will result.

⍱ Nor: Returns 1 if both right and left operands are 0, 0 if either one or both are 1. If operands are not logical an error will result.

Dyadic Mixed Operators

? Deal: Generates the number of random numbers specified by the left operand, between 1 and the right operand, such that there is no repetition of the numbers.

⍳ Index of: Returns the subscript to the left operand where the first occurrence of the right operand can be found. If the right operand is not found in the left operand then one plus the length of the left operand is returned instead.

, Catenation: Returns the two operands joined together as one vector (or array).

Another type of constant is the character type. 'A' is a character scalar constant and 'ABCD' is a character vector. A character vector in APL is roughly equivalent to the character string as implemented in languages like Extended BASIC, PL/1 or XPL.

Any legal left operand can be subscripted by using the brackets.

```
33 443 ¯34 [1 3]                USER ──►
```

gives a two element array

```
33 ¯34                          ◄── APL
```

In this case a three element array is reduced to two by choosing elements 1 and 3.

A variable, whose name can be up to 31 characters long, is assigned a value by the assignment arrow.

For example:

```
      D← 0 0 9                  USER ──►
      A← 1 2 3 4                USER ──►
      B← 5 6 7 ¯4               USER ──►
      C← A+B                    USER ──►
      D [1 2]← C[3 4]           USER ──►
      C                         USER ──►
6 8 10 0                        ◄── APL
      D                         USER ──►
10 0 9                          ◄── APL
```

Notice that an assignment suppresses the printing of a value. In this example D is given the vector value 0 0 9. A is given the vector value 1 2 3 4, and B is given the value 5 6 7 ¯4; C is the sum of these two vectors. The first two elements of D are assigned the last two of C. Character data can also be assigned to a variable, such as

```
      MESS←'This is a message'USER ──►
      MESS                      USER ──►
This is a message               ◄── APL
```

Some of the operators listed in table 1 are very important to APL programs. One of these is the monadic iota, known as the index generator, which creates a vector of consecutive integers from 1 to the value of the operand. This operator can be used to eliminate many loops from programs. For

example, if you wanted to calculate the cosine of a number using the Taylor series:

$$1 - \frac{x^2}{2!} + \frac{x^4}{4!} + \ldots \frac{(-1)^{n-1}x^{2n-2}}{(2n-2)\,!}$$

(instead of using the 2○X function) this expression would do it, where N is the number of terms to be calculated.

```
      +/ (¯1*¯1 +ιN) x (X*          USER ——→
      2x¯1+ιN)÷ ! 2x¯1+ιN           USER ——→
```

The subexpression `2x¯1+ιN` will generate N even integers, while this same expression is used to calculate factorials. The powers and the factorials are divided in parallel and multiplied by the `1 ¯1 1 ¯1 ...` generated by `(¯1*¯1+ιN)` The +/, called a plus reduction, sums up each one of the elements of the vector which contained the quotients of the powers and factorials. Another example of plus reduction would be to average the vector 2 3 4 5:

```
      (+/2 3 4 5) ÷ 4               USER ——→
3.5
```

Any other dyadic scaler operator could have been used; for example, instead of the built-in factorial operator, the factorial of a scalar J could be calculated by:

```
      x/ J                          USER ——→
```

Two other operators that eliminate loops are the grade up and grade down operators, which function similarly. They are monadic only. The grade up gives a vector of the subscripts to the given vector so that if the vector is subscripted by its grade up it will be sorted in ascending order. For example:

```
      A← 6 4 7 3 5                  USER ——→
      ⍋ A                           USER ——→
4 2 5 1 3                           ←—— APL
      A [⍋A]                        USER ——→
3 4 5 6 7                           ←—— APL
```

Suppose the scores of the players on your bowling team were in a vector subscripted by player number (as in previous examples); then if you wanted to display the ranking of players, best to worst, simply do a grade down: `⍒A`.

One of the most interesting operators of APL is the execute operator, sometimes represented as ⍎ or ε which takes a character vector and performs operations as if the characters had been input during evaluated input.

Table 1, continued:

ρ	Reshape: Returns a vector (or array) created from the elements of the right operand, such that the shape of this new vector equals the left operand.
⊥	Decode: Returns the base value, where the left operand is a vector of base weights, and the right operand is a vector of base representation. For example: 10 10 10⊥ 2 3 4 is 234 and 2 2 2 ⊥1 0 1 is 5.
⊤	Encode: Returns base representation as a vector, where the left operand is a vector of base weights, and the right operand is the base value. For example: 10 10 10⊤ 234 is 2 3 4 and 2 2 2⊤5 is 1 0 1.
↑	Take: If the left operand is positive, N↑A returns the first N elements of A. If the left operand is negative, (−N)↑A returns the last N elements of A.
↓	Drop: If the left operand is positive, N↓A returns a vector with the first N elements of the right operand removed. If the left operand is negative (−N)↓A returns the last N elements dropped.
ε	Membership: For each element of the left operand, membership returns a 1 if the element is contained in the right operand, 0 otherwise.
⌹	Matrix divide: Returns the solution of simultaneous linear equations, where the left operand is the vector of the constants, and the right operand is the matrix of the coefficients.
⍉	Dyadic transpose: Often used to find the diagonal of a matrix.
⌽	Rotation: Returns the right operand rotated the number of elements specified by the left operand (to the right if the right operand is negative, to the left if the right operand is positive).
/	Compression: Returns a vector (or array) where the elements of this vector were selected from the right operand when the corresponding element of the left operand was 1. If the left operand is not logical an error results. The length of the left operand must be equal to the length of the right operand; but it does not entirely obey the rule for scalar operators, since the result will have a length which is equal to the number of 1s in the left operand.
\	Expansion: Returns a vector (or array) with zeros inserted in the corresponding positions of the right operand wherever there is a zero in the left operand.

Composite Operators
(where f and g are any dyadic scalar operator)

f/	Reduction: Performs the operation on the elements of a vector (or array), right to left.
f\	Scan: Performs the operation on the elements of the right operand, left to right, and saves the results in a vector, which is what it returns.
f.g	Generalized matrix product: A+.xB is normal matrix multiplication.
o.f	Outer product: Can be used to generate multiplication tables, etc.

Special Operators and Characters

←	Assignment: Stores the expression on the right in the variable on the left.
⎕	Quad: Requests an expression for input and returns the value of the expression. Also used for output.
⍞	Quote-quad: Requests a character value to be input.
⍑	T-bar: For information about variable type, and character generation.
⌶	I-beam: Returns system dependent information.
⍝	Comment: APL ignores all characters to the right.
∇	Del: Used to open and close functions.
→	Branch: Continues execution of a function at the line whose number is equal to the value of the expression on the right. Branch to empty value branches to next line, branch to 0 stops the function.

For example:

```
      ⍎'1+1'                USER ⟶
2                           ⟵ APL
      A←'1'                 USER ⟶
      B←'1'                 USER ⟶
      ⍎A,'+',B              USER ⟶
2                           ⟵ APL
```

The comma is the catenation operator, which takes the vectors and forms a new vector of the old two joined together, so '+',B forms '+1' and A,'+1' forms '1+1'. This is then evaluated by the execute operator which gives the numerical result of 2. Catenation can also be used on numeric vectors:

```
      A←1 2 3               USER ⟶
      B←5.5 4.7 3           USER ⟶
      A,B                   USER ⟶
1 2 3 5.5 4.7 3             ⟵ APL
```

To create an APL program (function) you must leave the executive or evaluated input mode, and enter the function definition mode. To do this, type the del (∇) and the name of the function you wish to create. For example, to create a function called NEW type:

```
      ∇NEW                  USER ⟶
```

to which APL would respond:

```
[1]                         ⟵ APL
```

You would then type the first line of the function, and continue until all the function is entered, or until you want to leave the function definition mode, which you can do by typing another del. Here is a sample function:

```
      ∇NEW                  USER ⟶
[1]                         ⟵ APL
      N←0                   USER ⟶
[2]                         ⟵ APL
      N←N+1                 USER ⟶
[3]                         ⟵ APL
      N                     USER ⟶
[4]                         ⟵ APL
      → (N≠10)/2            USER ⟶
[5]                         ⟵ APL
      'FINISHED'            USER ⟶
```

To run this function, simply type its name: NEW.

This *function starts* at line 1 by initializing the counter N to 0, then it proceeds to line 2 where the counter is incremented. At line 4, N is compared to 10, and since they are not equal a result of 1 is produced, which allows the 2 to pass through the compression operator (see table 1). The → is the symbol for branching, so branch to 2 in the above example takes control back to line 2, where N is incremented again. The loop continues until N=10, in which case N≠10 returns a value of 0. This causes the compression operator to return a null value. Branching to a null value is the same as branching to the next line in the function, which in this example prints out FINISHED. Since there are no more lines to execute, execution of the function stops, and APL prints out six spaces like normal. The output of this function would look like this:

```
      NEW                   USER ⟶
1
2
3
4
5
6                           ⟵ APL
7
8
9
10
FINISHED
```

This program is not a very useful one, since the iota could be used to get essentially the same result, but suppose you wanted to change this function to output a table of numbers and their squares. First you would reenter the function definition mode, and type [3] to say that you wanted to change line three. You would type in:

```
      [3]N;' ';N*2          USER ⟶
```

The semicolon separates things to be printed on the same line. You decide you want a header on the output, so you type [1.5] to insert a line between 1 and 2. Then you list the function by typing ⎕ (the box is called a quad, and is also used for input and output within an expression).

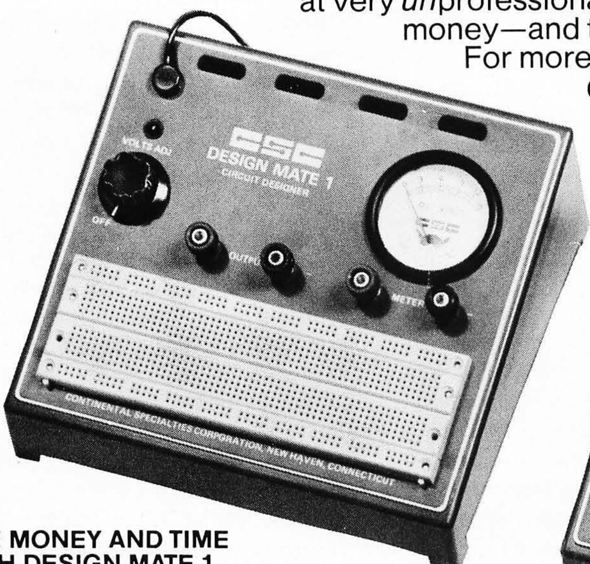
```
[1] N←0                     ⟵ APL
[1.5] 'TABLE OF SQUARES'    ⟵ APL
[2] N←N+1                   ⟵ APL
[3] N;N*2                   ⟵ APL
[4] →(N≠10)/2               ⟵ APL
```

Everything looks OK, so you close the

Continued on page 123

HOW TO SAVE HUNDREDS OF DOLLARS ON PARTS. PAINLESSLY.

At CSC, we've developed a family of ingenious Design Mate™ test equipment that gives you professional quality and precision at very *un*professional prices. Each unit can save you money—and time—in a number of interesting ways. For more information on these, or any other CSC products, see your dealer or write for our catalog and distributor list.





SAVE MONEY AND TIME WITH DESIGN MATE 1

This precision all-in-one unit combines a solderless plug-in breadboarding system with a built-in better-than-1%-regulated variable 5-15V supply and 0-15V voltmeter. Gives you everything you need to design and test circuits faster than you ever could before. Saves money by eliminating lead damage and heat damage to components. Lets you re-use parts over and over again, to save even more. All for just $54.95*



STRETCH YOUR BUDGET FURTHER WITH DESIGN MATE 2

Precision function generator lets you test all kinds of equipment, with 1Hz-100kHz signals. Low-distortion sine waves, high-linearity triangle waves, fast-rise-time square waves. Five decade ranges, accurate to 5% of dial setting, with variable 100mV-10V P-P output and constant 600-ohm impedance. At $69.95* it's a lot of signal for very little money.

SAVE MORE MONEY AND TIME WITH DESIGN MATE 3

Accurate R/C bridge helps you use "bargain" components. Quickly and easily measures resistance 10 ohms—10 meg; capacitance 10pF-1μF—both in decade ranges to within 5% of dial setting. Simple, 2-control operation and positive LED indication make measurements in seconds. At $59.95* it pays for itself in no time.

CONTINENTAL SPECIALTIES CORPORATION

CSC

EASY DOES IT

44 Kendall Street, Box 1942
New Haven, CT 06509 • 203-624-3103 TWX: 710-465-1227
West Coast office: Box 7809, San Francisco, CA 94119 • 415-421-8872 TWX: 910-372-7992
Canada: Len Finkler Ltd., Ontario

*Manufacturer's suggested list. Prices and specifications subject to change without notice.

Build the Beer Budget

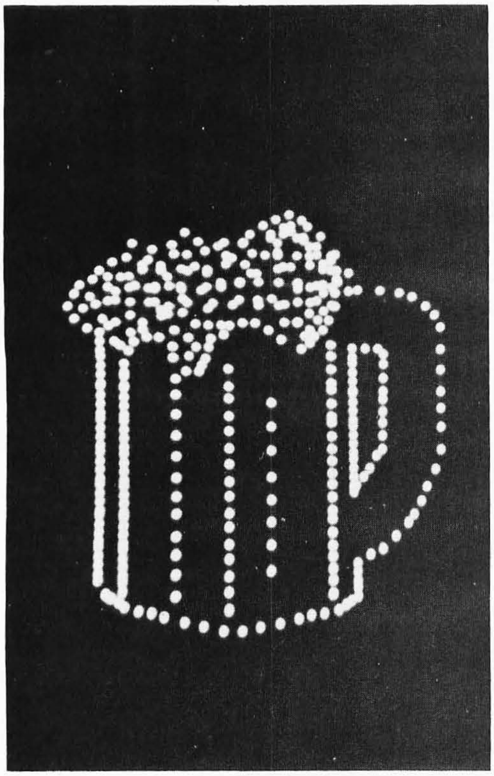


Photo 1: This image was inspired by the article's title (or was it the other way around?). The program of listing 1 was used to draw this picture as well as the others accompanying this article.

Peter Nelson
68 Seaver St
Wellesley MA 02181

In the past, graphics systems available to the experimenter have been expensive and complicated or have forced him to settle for limited resolution. The simple circuit described in this article will provide resolution of 128 by 128 dots for under $20 and four times that resolution for just over $20. It requires the use of an oscilloscope as the display device, but it need not use an expensive scope.

We can create images on the screen of an oscilloscope by using dots. By placing many dots close together, we can form lines and shapes or even letters and numbers. The location of a dot on the screen is specified to the oscilloscope by a pair of voltages, one voltage being applied to the scope's vertical input and the other to its horizontal input. The computer provides the voltage values by outputting two binary words to a pair of digital to analog converters (known as DACs). The converters produce output voltages that are proportional to the numeric value of the words. Because an 8 bit word will allow the DAC to produce any one of 256 voltage values, it should be possible, using high precision 8 bit DACs, to position a dot anywhere in a 256 by 256 field.

By providing a rapid sequence of many binary word pairs, we can generate a large number of dots that all appear to exist on the screen at the same time. By repeating the sequence many times a second, we can display a steady, nonflickering pattern of dots. Each time we repeat the sequence we "refresh" the display.

Because the time required for the computer to perform a refresh operation will depend, at least in part, on the number of dots being displayed, there will certainly be an upper limit to the total number of dots that we can display without a noticeable flickering of the screen. That limit is determined by such factors as the speed of the computer, what the computer is doing besides refreshing the display, how the numbers are obtained (ie: whether they are computed or read from a buffer in memory), the persistence of the oscilloscope's phosphor, and what flicker frequency the user considers noticeable. For example, using the routine described later in this article on an 8080 system with a 1 MHz clock (about half the speed of an Altair 8800) and using a medium persistence phosphor in the display, I find that I can display about 512 dots before flicker begins to appear.

The Circuit

The circuit of figure 1 uses components that are all readily available. Most can be bought through the suppliers who advertise in the back pages of BYTE magazine. The MC1408 digital to analog converters can be supplied by any Motorola distributor and are manufactured in several different levels of precision. The precision determines the exactness with which the output voltage will correspond to the binary input. The -L8

Table 1: Integrated Circuit Power List. This list contains the pin connections for the power voltages used by figure 1.

Integrated Circuit	Type	+5 V Power	Ground	−V (−7.1)
IC1	74100	24	7	–
IC2	74100	24	7	–
IC3	MC1408L7 or MC1408L8	13	2	3
IC4	74100	24	7	–
IC5	MC1408L7 or MC1408L8	13	2	3
IC6	7404	14	7	–

Graphics Interface

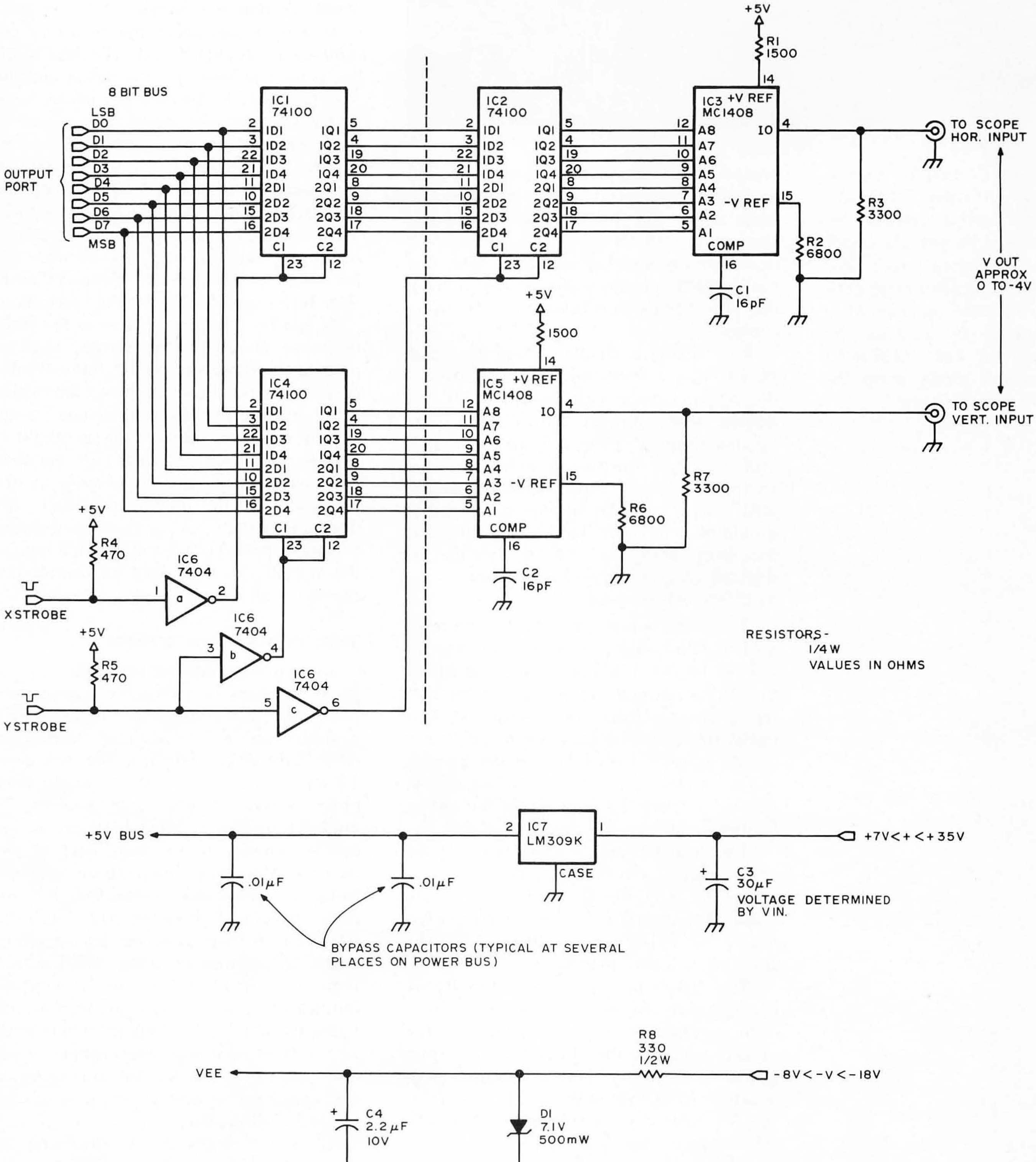


Figure 1: Schematic for the Beer Budget Graphics System. This graphics system uses the computer's main memory as a refresh source, with a continuously executing program to put the position information out to the X and Y inputs of the oscilloscope. Both the X and Y outputs change "simultaneously," limited only by the amplifier slew rates of the oscilloscope and the MC1408 settling times. The driver program enables interrupts once per scan in order to allow a keyboard controlled drawing mode commanded by user inputs.

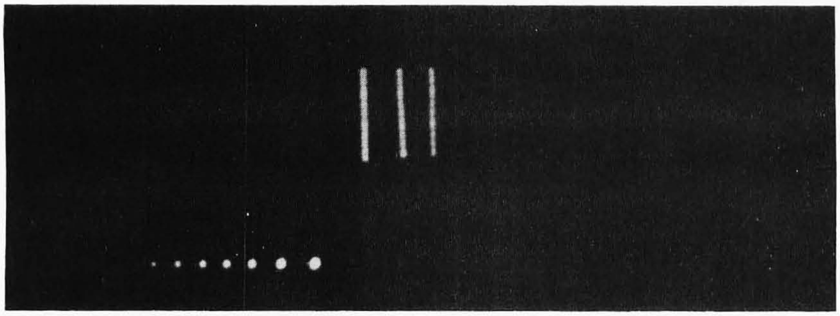


Photo 2: An illustration of how different dot brightnesses are achieved by repeating points. The cursor is a winking point. For user controlled brightness, extra time on a point is entered by pushing the "deposit key" (ASCII'') multiple times using the program of listing 1.*

version should be used if a 256 by 256 resolution is desired. The -L7 version may be used for a 128 by 128 resolution, but attempts to use it for higher resolutions will result in unevenly spaced dots. For low resolution applications (64 by 64 or less), the -L6 version provides an economical choice.

This circuit is simple enough to breadboard using a socketboard array such as I did. Later, I made a permanent board as a double layer printed circuit with fake "plated through" holes of wire jumpers. It may also be constructed using wirewrap techniques. Leads, especially those associated with the digital to analog converters, should be kept short; and I recommend that the lines going to the oscilloscope be shielded. A good ground connection to the oscilloscope is essential.

The data bus is assumed to come from an 8 bit parallel output port that can sink 6.4 mA per bit line. I do not recommend driving this circuit without additional current buffering directly from the system data bus unless there are relatively few other devices on the bus. A dotted line in the drawing shows where a 17 line interface to an existing latched IO port might be made, further simplifying the circuit.

The strobe lines are active low and the entire logic 0 transition should take place while the data inputs to the latches are stable. The inverters provide current buffering for the 74100 clock inputs as well as the appropriate logic 1 load signal.

The digital to analog converters in the configuration shown in the schematic produce a negative output voltage. As the numeric value of the binary word becomes greater, the output voltage becomes more negative. Note that with the MC1408 DAC the A8 input is the least significant bit and A1 is the most significant bit.

Choice of an oscilloscope is not very critical. I have been using a battle-scarred Heathkit IO-102, which, until they stopped producing it, was the bottom of their line. The oscilloscope must have both X and Y inputs. If you have direct-coupled inputs, you may find them preferable at times, though for most applications AC or DC coupled inputs can be used. Unless it is severely limited, bandwidth should not be a problem: 1 MHz or greater in both channels should be more than adequate.

In normal operation, the horizontal coordinate is outputted first. The XSTROBE line is used to load that coordinate into the first latch of the horizontal channel. Note that it does not yet appear at the input of the horizontal digital to analog converter. The vertical coordinate is then outputted and loaded into the vertical latch by the YSTROBE line. This strobe signal also loads the horizontal coordinate into the second horizontal latch. Thus, the vertical and horizontal coordinates are both applied to their respective converters at the same time.

If you find that the dots seem to vibrate or move about on the screen, you are probably picking up 60 Hz hum. Double check the ground connection to the oscilloscope and use shielded cable if you are not already doing so. Streaked or smeared dots suggest excessive reactance on the lines going to the oscilloscope or severely limited bandwidth of the oscilloscope itself. The direction of the smearing gives you a clue as to which channel is being affected. Reducing the refresh rate may help to control this problem if all else fails.

Some Software Considerations

In many graphics applications such as game programs or interactive graphics, the computer is not merely being used to refresh a display but is also called on to perform other functions, in effect, at the same time. To allow maximum freedom for the computer to execute these other functions, the display refresh routine should be designed to operate quickly rather than, say, to save memory. Rewriting a dot output routine so as to save a few microseconds may not be a petty exercise if there are 512 dots to be refreshed 20 times a second. Such a savings might be obtained by using JUMP instructions instead of CALLs or, if you are outputting from a buffer in memory, defining the limits of that buffer only in terms of the most significant address byte so you don't have to perform a time-consuming 16 bit comparison to find out if you've reached the end of the buffer.

It is also important to distribute the display refreshes evenly to avoid annoying flicker of the screen. If you decide, for example, to refresh the display 30 times a second, it will do no good to perform all 30 refresh operations in the first half of that second. This would probably be a good application for a programmable interval

timer which would generate an interrupt when it is time to refresh the display.

Note that it is possible to use the software to control dot brightness. By causing a particular point on the screen to be refreshed more often than other points, we can generate a brighter dot at that point. This may be done as simply as storing that dot in several locations in the buffer.

The routine detailed in listing 1 was the first program that I wrote for the graphics circuit. It is written for use with an 8080 instruction set and is basically a picture drawing routine. It has a 512 dot buffer, a winking cursor, and it recognizes six commands: move cursor right; move cursor left; move cursor down; move cursor up; deposit a dot (at the cursor location); and, clear display. Because this routine uses a fixed buffer size, the clear is performed by sweeping all 512 dots into a pile in the lower left hand corner of the display (coordinate: FF,FF). Each command is indicated with a single ASCII character inputted from an interrupt driven keyboard. The interrupt service routine merely loads the accumulator from the input port and returns to where it was called from. Because I use "memory mapped" IO (ie: my IO ports are treated as memory locations), I use STA as an output instruction. Those using 8080 IO instructions will want to substitute OUT instructions.

Consider what the routine has to keep track of. When refreshing the display, we have a pointer (HL) that tells us which dot in the buffer we are refreshing, but when depositing a dot we have to know where in the buffer we have space for a "new dot" (HL on stack). We also have a screen location pointer (DE) for the cursor and a counter (C) to determine the cursor's winking rate. If a routine like this is used as a subroutine, care should be taken to save everything on the stack before returning to the main program.

Listing 1: An Intel 8080 program to drive the Beer Budget Graphics system. This program uses inputs from an ASCII encoded keyboard to command various cursor movements and perform the utility functions of clearing the screen and entering data. The cursor is an extremely bright point whose coordinates are maintained in the D and E registers. Data is kept in locations 300 to 6FF.

Addr	Hex Code			Label	Op	Operands	Commentary
0200	31	F0	0F	START	LXI	SP,FOOF	Initialize stack pointer;
0203	21	00	03		LXI	H,300	Initialize next new dot pointer
0206	E5				PUSH	H	and push onto stack;
0207	21	00	03	REFRESH	LXI	H,300	Set refresh pointer to start of buffer;
020A	7E			MORE	MOV	A,M	A := @HL [get horizontal coordinate];
020B	32	05	80		STA	OUTPUTX	X coordinate := A [memory location is output];
020E	23				INX	H	HL := HL + 1 [point to the vertical coordinate];
020F	7E				MOV	A,M	A := @HL [get vertical coordinate];
0210	32	06	80		STA	OUTPUTY	Y coordinate := A [memory location is output];
0213	23				INX	H	HL := HL + 1 [point to horizontal coordinate];
0214	3E	07			MVI	A,7	A := buffer page address limit;
0216	BC				CMP	H	Is HL pair at end of buffer?
*0217	C2	0A	02		JNZ	MORE	If not then repeat the display loop;
021A	0D				DCR	C	C := C − 1 [decrement cursor counter];
*021B	CA	77	02		JZ	CURSOR	If zero then output the cursor pattern;
021E	AF			INPUT	XOR	A	A := 0;
021F	FB				EI		Enable interrupts from keyboard;
0220	00				NOP		Interrupt window to service keystroke;
0221	F3				DI		Disable interrupts from keyboard;
0222	FE	00			CPI	0	Was there any input?
*0224	CA	07	02		JZ	REFRESH	If not then go refresh display again;
0227	E6	7F			ANI	7F	A := A & 7F [remove most significant bit with mask];
0229	FE	4C			CPI	'L'	Was it move cursor left?
*022B	CA	4A	02		JZ	LEFT	If so then go move the cursor left;
022E	FE	52			CPI	'R'	Was it move cursor right?
*0230	CA	4F	02		JZ	RIGHT	If so then go move the cursor right;
0233	FE	55			CPI	'U'	Was it move cursor up?
*0235	CA	54	02		JZ	UP	If so then go move the cursor up;
0238	FE	44			CPI	'D'	Was it move cursor down?
*023A	CA	59	02		JZ	DOWN	If so then go move the cursor down;
023D	FE	2F			CPI	'/'	Was it a clear command?
*023F	CA	5E	02		JZ	CLEAR	If so then go clear the display;
0242	FE	2A			CPI	'*'	Was it a deposit dot command?
*0244	CA	6E	02		JZ	DEPOSIT	If so then go deposit a dot;
*0247	C3	07	02		JMP	REFRESH	Bad command, so back to refreshments;
024A	14			LEFT	INR	D	Move cursor left one space;
024B	14				INR	D	[Second move, NOP if 8 bit DAC used];
*024C	C3	07	02		JMP	REFRESH	Refresh the display;
024F	15			RIGHT	DCR	D	Move cursor right one space;
0250	15				DCR	D	[Second move, NOP if 8 bit DAC used];
*0251	C3	07	02		JMP	REFRESH	Refresh the display;
0254	1D			UP	DCR	E	Move cursor up one space;
0255	1D				DCR	E	[Second move, NOP if 8 bit DAC is used];
*0256	C3	07	02		JMP	REFRESH	Refresh the display;
0259	1C			DOWN	INR	E	Move cursor down one space;
025A	1C				INR	E	[Second move, NOP if 8 bit DAC];
*025B	C3	07	02		JMP	REFRESH	Refresh the display;
025E	21	00	03	CLEAR	LXI	H,300	HL := starting address, refresh buffer;
0261	3E	FF		AGAIN	MVI	A,FF	A := null coordinate value, lower left;
0263	77				MOV	M,A	@HL := A [zap a buffer location];
0264	23				INX	H	HL := HL + 1 [point to next location];
0265	3E	07			MVI	A,07	A := page address of buffer limit;
0267	BC				CMP	H	Is HL at the limit?
*0268	C2	61	02		JNZ	AGAIN	If not then continue zapping memory;
*026B	C3	00	02		JMP	START	When done, restart the program;
026E	E1			DEPOSIT	POP	H	Pull old buffer location pointer;
026F	72				MOV	M,D	@HL := horizontal cursor position;
0270	23				INX	H	HL := HL + 1 [point to next coordinate in buffer];
0271	73				MOV	M,E	@HL := vertical cursor position;
0272	23				INX	H	HL := HL + 1 [point to next coordinate];
0273	E5				PUSH	H	Push current buffer location pointer;
*0274	C3	07	02		JMP	REFRESH	Refresh the display;
0277	7A			CURSOR	MOV	A,D	A := horizontal position;
0278	32	05	80		STA	OUTPUTX	X coordinate := A;
027B	7B				MOV	A,E	A := vertical position;
027C	32	06	80		STA	OUTPUTY	Y coordinate := A;
027F	0E	08			MVI	C,8	Set cursor counter to 8;
*0281	C3	1E	02		JMP	INPUT	Go get more inputs;

Notes:

All constants are hexadecimal;

@HL in commentary indicates the contents of memory location pointed to by HL pair;

*next to address indicates the instruction contains an address which must be recomputed if the program is relocated;

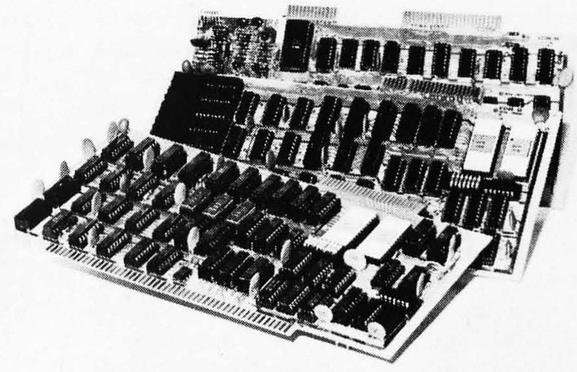
'X' indicates ASCII character X;

Note that 16 bit data format for the 8080 stores the low order byte in the lowest address, thus the hexadecimal value 8006 (for example) is seen in the listing as 06 80.

For 8 bit converters, the increment or decrement instructions at locations 024B, 0250, 0255, and 025A must be replaced by NOP instructions.

Other Ideas

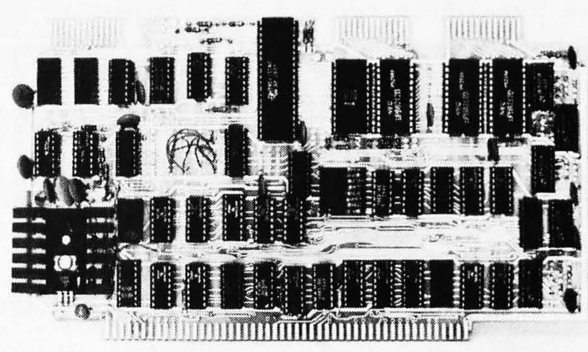
Can a pair of digital to analog converters connected to a computer be used for applications other than graphics? Absolutely! An obvious idea would be waveform generation, but how about musical instrument synthesis? One converter could be connected to a wide range voltage to frequency converter such as the Raytheon 4151 and the other converter could provide envelope information. Passive components could tailor the output of the 4151 to suit the instrument being synthesized. Just about any continuous range phenomenon can be put under the control of the computer using a digital to analog converter. The Motorola application sheet for the MC1408 includes ideas such as programmable gain amplifiers, a programmable power supply, a programmable constant current source, and others.■



There's another side of IMSAI you should know

IMSAI VIDEO DISPLAY
Assembled $1199.00/Unassembled $699.00

IMSAI MULTIPLE I/O BOARD
Assembled $350.00/Unassembled $195.00



Sure you know our IMSAI 8080 computer. Its size, quality, performance and price set the industry standard for value. There's another, equally important side of IMSAI—our broad and ever growing I/O product line. Right now IMSAI offers an array of sensibly priced, versatile, high-quality interfaces and peripherals that live up to the IMSAI 8080 for quality, technical leadership, and value.

IMSAI VIDEO DISPLAY. We call it the most powerful, flexible video display interface available. With up to 3 boards that fit into your IMSAI you can display color characters and color graphics. simultaneously. · Unlimited image control. · Large capacity display. · Eight colors; 7 half tones. · All cables included. · Full ASCII character set. · Character blink, reverse video, multiple cursor and double-width characters.

IMSAI KEYBOARD. 53-key, ASCII encoded keyboard with good touch quality and upper case alphanumeric keys that duplicate functions of the ASR-33. Supplied with commercial grade cabinet and cable. Interfaces to any 8-bit parallel port.

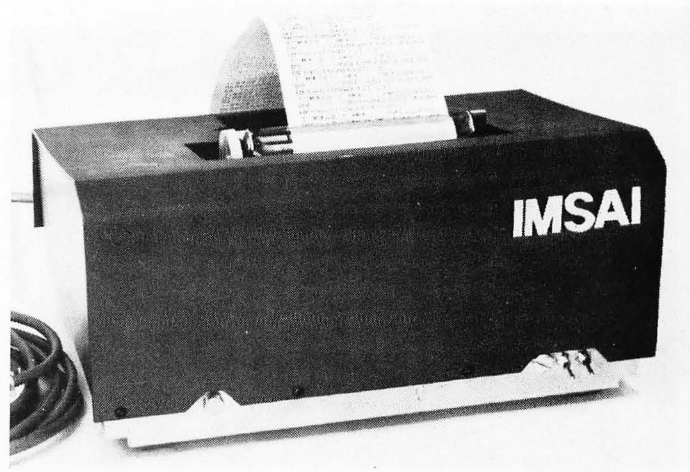
IMSAI 44 COLUMN PRINTER. 44 column dot matrix printer offering hard copy output at an affordable price. Interfaces to many computers using an 8-bit parallel output port. Simple programming and installation. Self-contained with case, cable, power supply, timing, control and character generation included. Produces 75/lines/min.; double size characters software selectable.

IMSAI MULTIPLE I/O BOARD. Simultaneous control of keyboard, printer, tape cassette (Byte or Tarbell) and teletype (or CRT) by a single MIO board. Our 5-port, multiple I/O board combines the most commonly used I/O interfaces including a cassette interface. faces including a cassette interface, two parallel I/O ports, one serial channel and one control port—all on one board!

And many other **IMSAI** I/O interfaces to expand your system. Examples:

IMSAI Programmable Parallel I/O board offering 6-ports for the ultimate in I/O flexibility and control.

IMSAI PRINTER
Assembled $549.00/Unassembled $399.00



IMSAI KEYBOARD
Assembled $199.00



IMSAI 2-Channel serial I/O board with 3 modes for total versatility in communications.

IMSAI 4-Port Parallel I/O board with LEDs for visual observation of data flow.

IMSAI Intelligent Breadboard for easier, more accurate circuit design and classroom instruction.

IMSAI 4K RAM—the best price performance 4K RAM board on the market.

Socket sets for all boards. Use for easier board maintenance.

For commercial application—Our IMSAI floppy disc with disc operating system and extended BASIC.

ALL these IMSAI products available NOW at your IMSAI dealer or from the factory.

Yes, the other side of IMSAI has much to offer. It's a complete I/O line with products that match the IMSAI 8080. We offer a detailed, illustrated catalog describing the IMSAI 8080, its options, peripherals, software, prices and specifications. Send one dollar to cover handling. Write or call for name of the nearest dealer.

Dealer inquiries invited.

IMSAI

IMS Associates, Inc.
14680 Wicks Boulevard
San Leandro, CA. 94577
(415) 483-2093
TWX: 910-366-7287

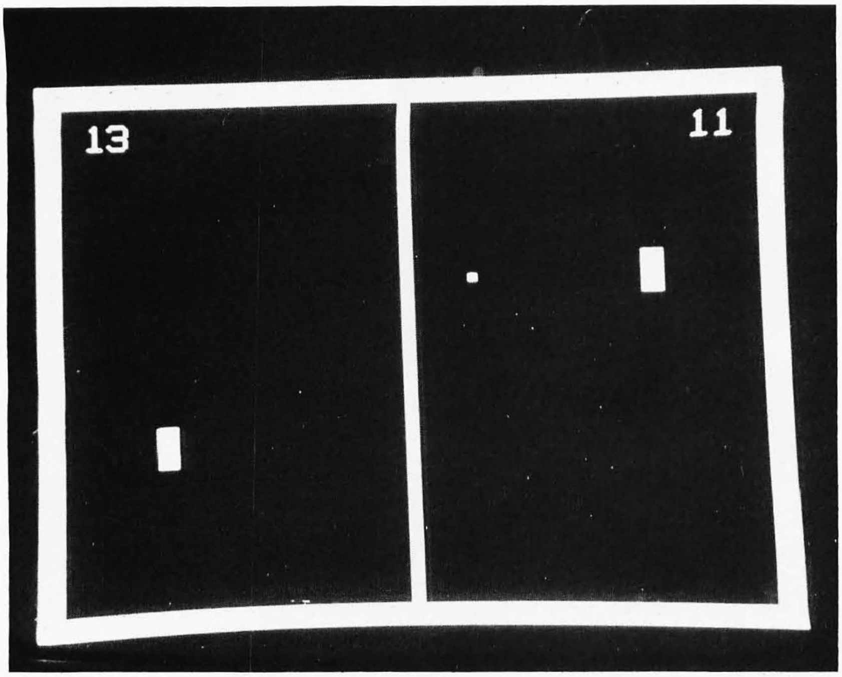


Photo 1: With a high resolution display of 256 by 208 points, the author's display system becomes a programmable blackboard for numerous games and visual outputs. A relatively trivial game is the simulated tennis often seen in bars, restaurants and airport waiting lounges (to say nothing of mass produced special purpose home video game sets). The software required is relatively simple and the display has only three moving objects – the two paddles and the ball. As in all interactive games, some form of player input is required. This can typically be supplied by a variable resistance (as in a potentiometer, slide pot or joystick) and an analog to digital converter.

Add This Graphics Display to Your System

Thomas R Buschbach
12310 Purcell Rd
Manassas VA 22110

Since one of the reasons I built a personal microprocessor was to play video games, I needed some type of high resolution graphics display capability. After examining the current unavailability of a low cost commercial product to perform this task, I decided to build my own unit. The main reason, besides cost, was that I wanted a 256 by 256 dot matrix (as you'll see below, I actually got 256 by 208). Since my system already contained a television display generator board for character information, I decided to build a graphics display unit which would utilize existing synchronization, blanking, and timing pulses. For storage of the data to be displayed, I started with a standard programmable memory board populated with 8 K eight bit bytes of 500 ns access time 2102 integrated circuits.

My design goals for this project were to:

- Utilize all possible circuitry existing on the television display interface board.
- Have the graphics display work in parallel with the existing alphanumeric character display.
- Have the ability to use the graphics display region of memory for ordinary programming purposes if the display is suppressed.
- Use direct memory access techniques (DMA) so that the graphics display data appears as ordinary programmable random access memory as far as the computer is concerned.

I was able to meet all these goals for a total cost of $25 in integrated circuits, above the cost of the existing display generator and memory boards of my system. If your system does not have an 8 K programmable memory board available, you will need to purchase one. Remember, when the graphics display is not in use, the memory can be utilized as if it were an ordinary programmable memory region; so the investment will also pay for itself in terms of raw memory power for your non graphics applications.

If your system does not have a television display generator board, the synchronization, blanking, and timing pulses can easily be generated using counters. I would suggest reading Don Lancaster's new book, *TV Typewriter Cookbook.* Chapter 4 contains

an explanation and schematics of the circuits to generate the required pulses.

The graphics display unit functions as an interface between the processor, the 8 K memory board, and the television display generator, to provide the graphics capability. A block diagram of this interface is shown in figure 1. The graphics display unit has two modes of operation, display mode and update mode. In display mode, the unit cyclically accesses all addresses of the programmable memory board using direct memory access, then converts the output from parallel to serial, sending it to the television display generator to be combined with any alphanumeric information and displayed on the monitor or TV set. Since this operation must occur in parallel with normal processor functions, the address bus of the memory must be isolated from the processor's address bus. The output data from the memory must also be isolated from the system's data buses. In the update mode, the

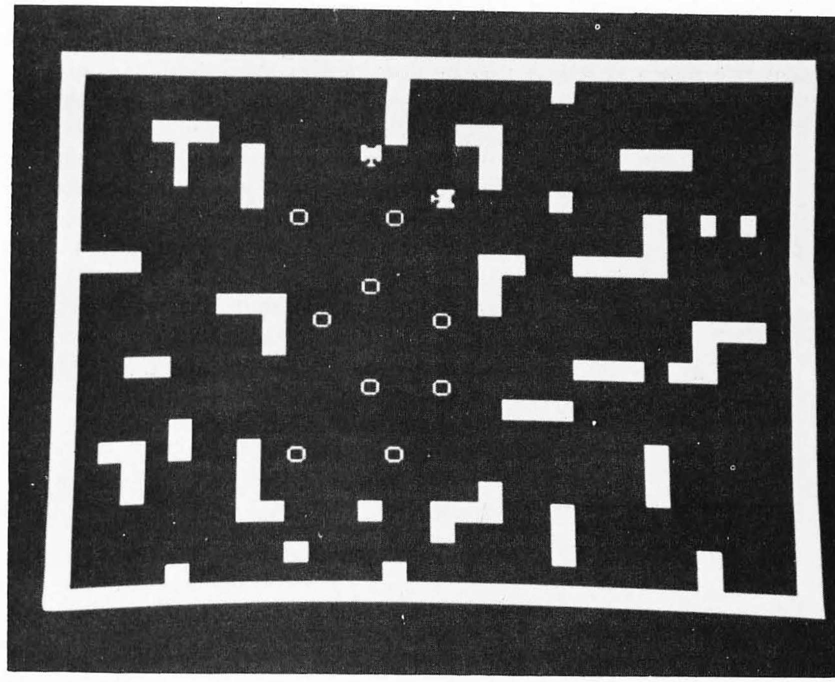


Photo 2: With a high resolution display, games need not be kept simple. Here is an example of a much more complex game display on the author's system, again modeled after the commercial games often found in entertainment centers. To set up such a display typically requires software to create the background maze, keep track of all the movable pieces, and of course program the logic of the game simulation.

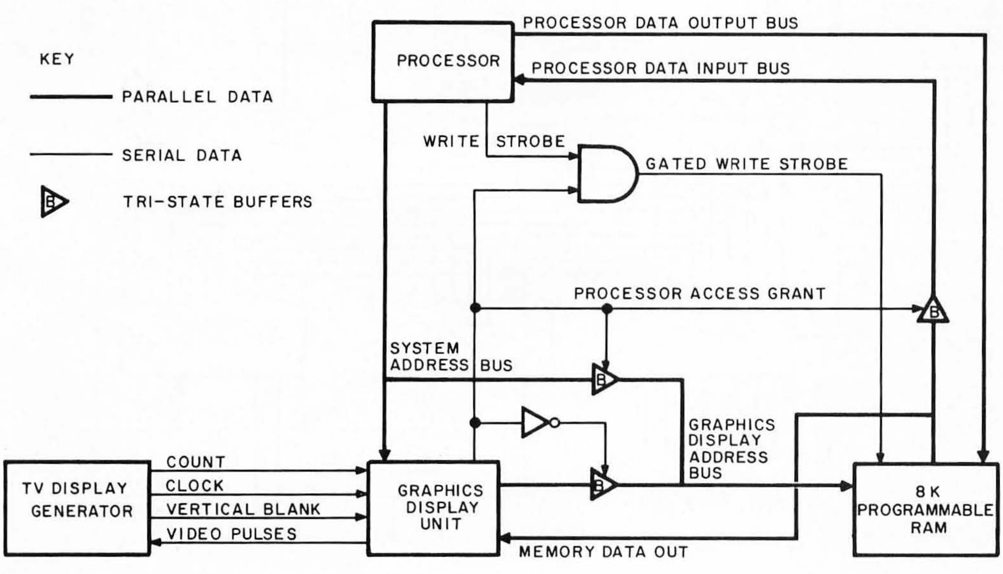


Figure 1: System Diagram. The graphics display unit interfaces to three elements of an existing system: The processor and its buses, the television display generator, and the 8 K programmable random access memory. The graphics display unit is normally in the display mode, where it continually displays data contained in the programmable memory module. If the processor references a location within the programmable memory's address space, the graphics display unit enters the update mode either immediately or during the next vertical blanking pulse depending upon the setting of a switch. In the update mode, the processor commands the memory and the display looks at random output bits, which are of course ignored during the blanking period.

processor is given control of the memory's address bus and allowed to read or write data. The update mode should only be entered while the television display generator is sending a vertical blanking pulse. This prevents the update from causing a spurious blip on the screen. However the penalty for observing this requirement is that the processor may have to wait for nearly a complete television frame time, about 13.5 ms maximum in the author's system. A memory wait this long is impossible for processors such as the 6800 which do not have a memory ready mechanism, in which case an alternate mechanism such as an input port bit line could be used.

A more detailed block diagram of the functions of the graphics display unit is shown in figure 2. This design is composed of three functionally distinct areas: the three state buffers and their control logic, the counters, and the shift registers. The block diagram and details of this article assume also that the processor has a split "in" and "out" memory data bus as is utilized in the Altair 8800 or Digital Group designs.

The state of the buffers is determined by the mode in which the graphics display unit is operating. In the display mode, the buffer, block (2) in the figure, allows signals to pass while buffers of blocks (1) and (3) must be in a high impedance state. This gives the graphics display unit's counters control over the programmable memory's address bus and blocks the output data from entering the processor's data input bus. In the update

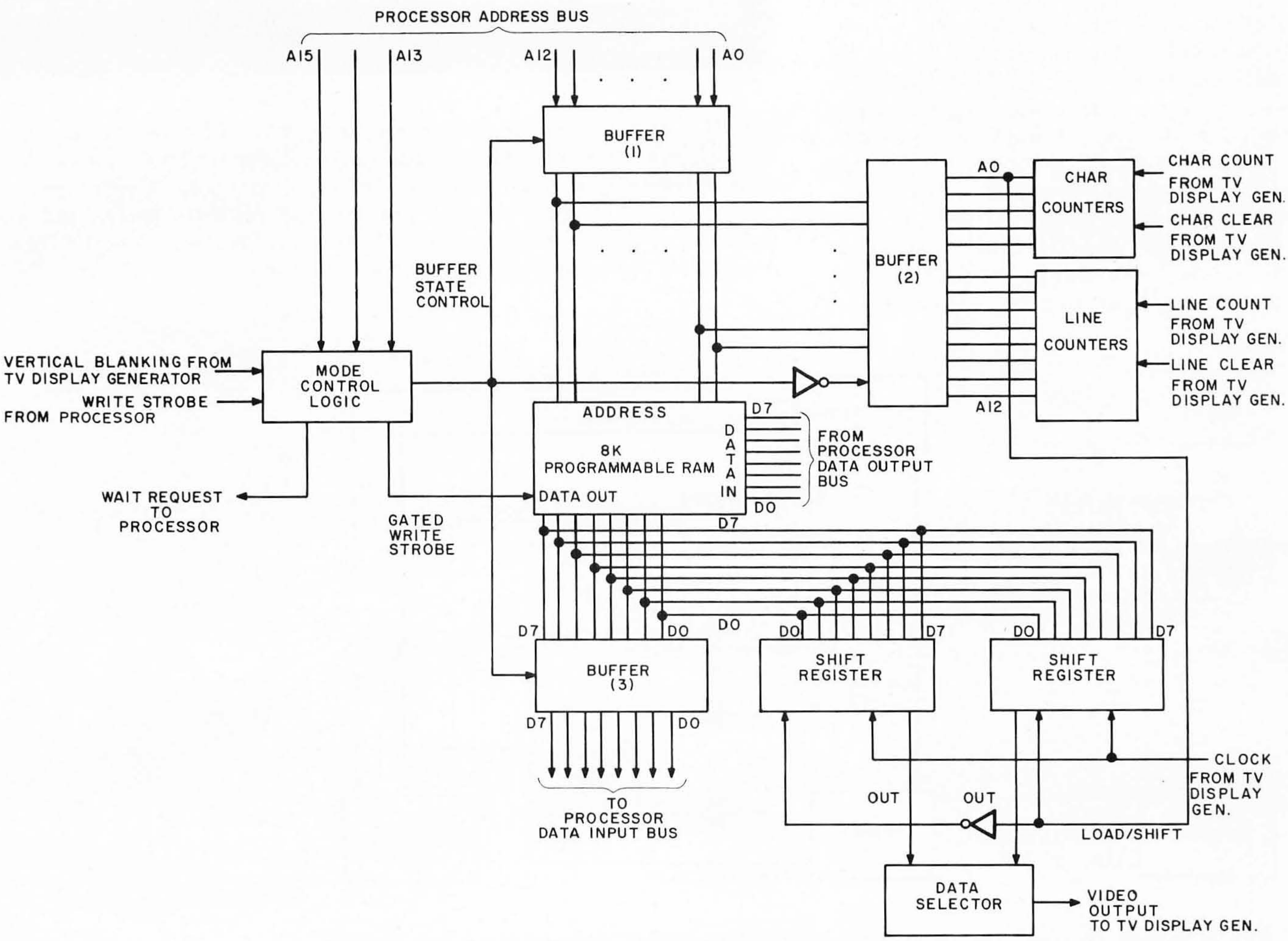


Figure 2: Detail Block Diagram. As in all "top down" system designs, the design progresses by filling in more and more detail. Here is a block diagram of the actual graphics display unit. This figure shows a detailed functional arrangement of the interface as constructed by the author. In the display mode, the graphics display unit continually cycles through all addresses in the programmable memory, outputting serial video levels to the television display generator, for mixing with existing character graphics information. In the update mode, the graphics display unit allows the microprocessor to control the programmable memory unit's address and data lines for normal access. The mode switching is automatic based upon whether or not the address sent by the processor is within the display's address space.

mode, the buffers (1) and (3) allow signals to pass while the buffer (2) is in a high impedance state. This gives the processor control of the programmable memory's address bus and allows memory to be connected to the processor's data input and address buses. The gated write strobe pulse is also enabled in this mode. Note that the processor is the only source of data written to the memory, so there is no switching of the processor data output bus which drives the inputs of the memory banks.

The mode control logic determines the proper mode of operation at any given time. It does this by monitoring the three most significant bits of the processor's address bus to determine if it is referencing a location in the programmable memory region assigned to the graphics display. If it is not referencing graphics memory, then the graphics display unit will remain in the display mode. If the processor does reference a location in the graphics display region of address space, then the television display generator is checked to see if it is generating a vertical blanking pulse. If it is, and if the graphics display is set (using S1) to prevent visual noise ("sparkle") caused by updates, then the display is changed into the update mode and the memory access takes place as if the processor were addressing an ordinary location. However if the television display generator is not in the vertical blanking period, and the graphics display is set via S1 to avoid arbitrary data during updates, then the processor is placed in a wait state until the next vertical blanking pulse is initiated, at which time the update mode is again entered. Note that this feature will only work with processors incorporating a memory ready line for slow response memories; an alternate mode of operation in which updating has priority over display must be used for processors which cannot wait.

The counters control the memory address bus during display operations, continually cycling through all locations with a full cycle taking 1/60th of a second (16.67 ms). These counters must be in synchronization with the existing counters in the television display generator. Most such television display generators contain two sets of counters. One set counts the character's position on the line. Each character is typically 8 bits wide, and there are typically 32 characters per line for a total of 256 dots per horizontal line. The other set of counters counts the number of horizontal lines which have been generated (TV scan lines, not character lines). In a typical television display without interlace there would be a maximum of 256 horizontal lines. On the television display generator

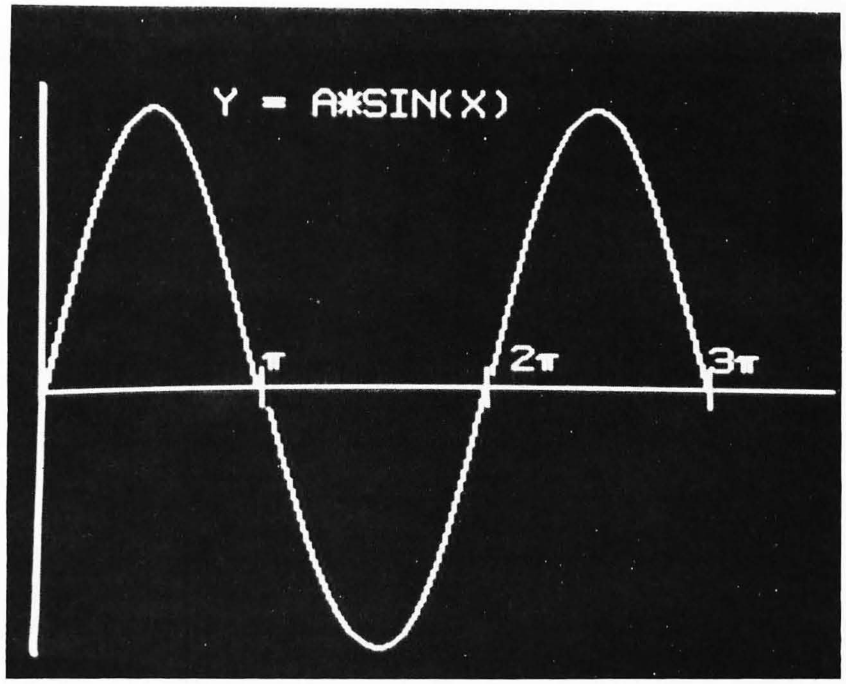
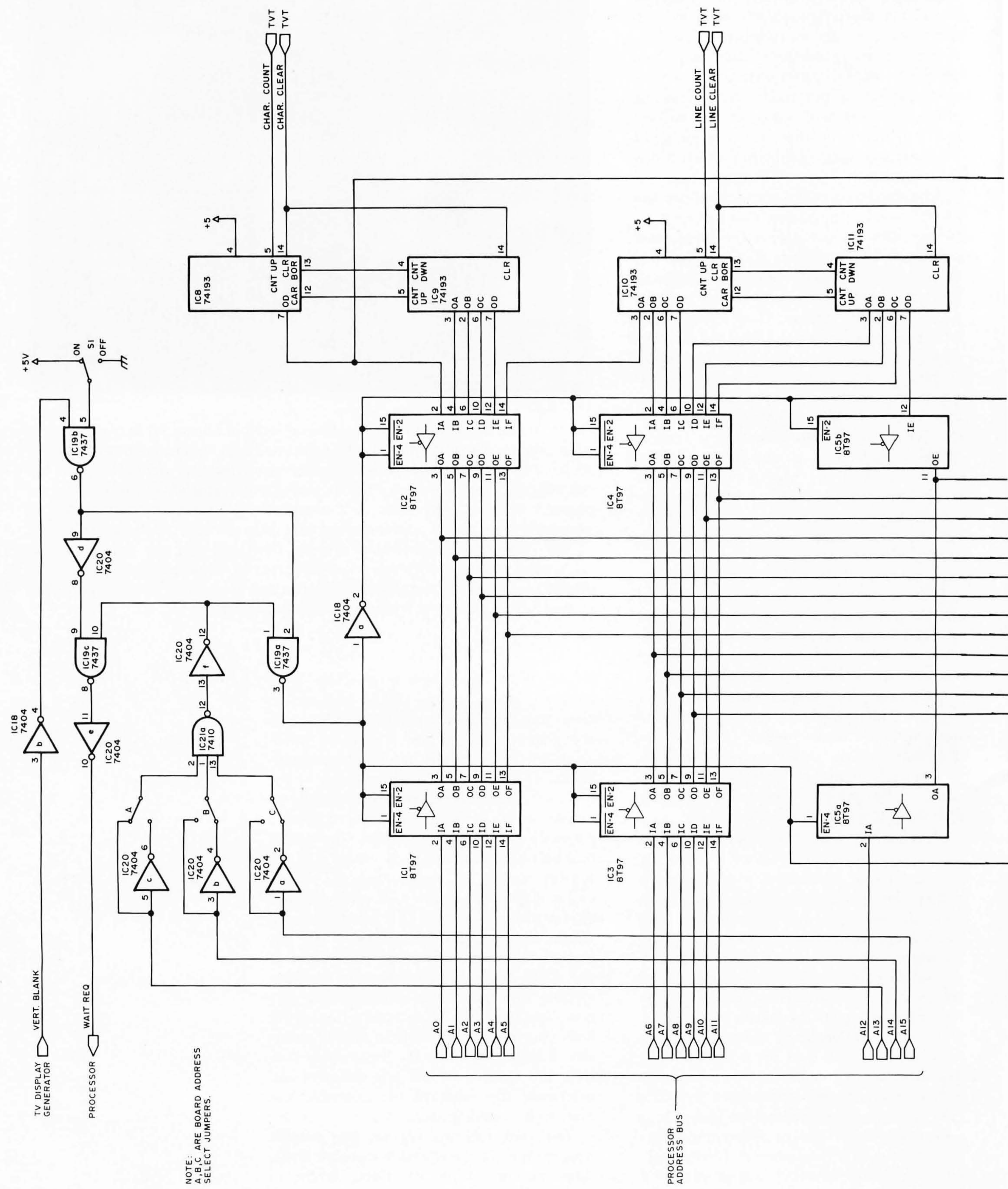


Photo 3: Not all outputs need be game oriented of course. Here is an example of a graphic output which could easily be used for engineering purposes: plotting the two dimensional behavior for some function. Here the author has calculated a sine wave using a BASIC package and appended various character graphics captions. The basic X-Y axes and the character graphics can be generated typically by a table lookup with some calculation; the actual points of the curve can be generated directly from the function calculation by calculating integer positions. In either case, to conveniently program the display, a set of bit manipulation subroutines is required to read and write individual bits, given the coordinates within the matrix.

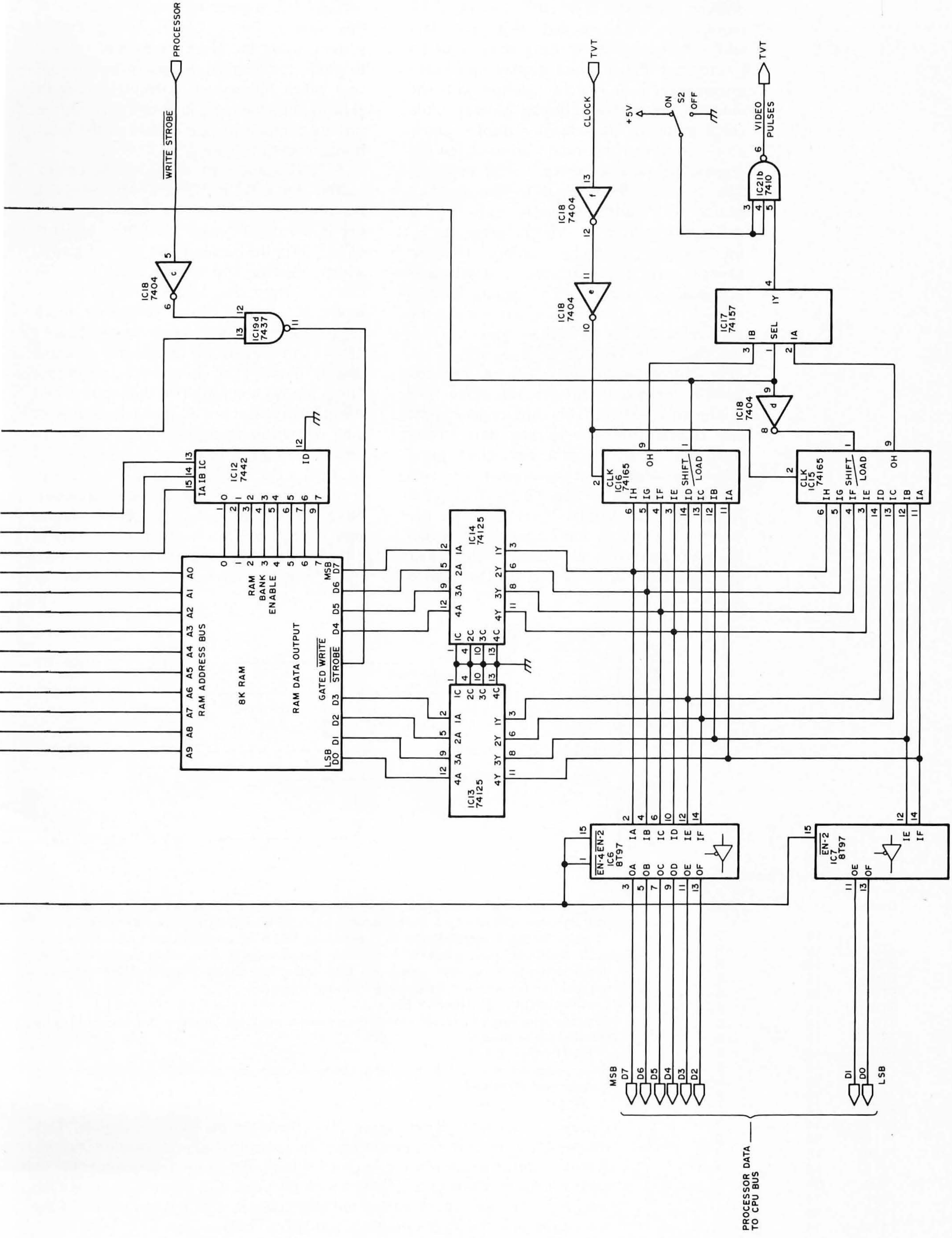
board, there are pulses which clear these counters. These pulses should be used to clear the corresponding set of counters in the graphics display unit. The pulse which clears the line counters has an added significance: It also determines how many of the possible 256 horizontal lines will actually be visible on the screen. Due to overscan on a conventional television screen, the actual number of lines visible is never 256. A typical number of visible lines for a television display generator such as my Digital Group product is 208. This means that the visible display is actually a 256 by 208 dot matrix. Note that this leaves 1536 bytes of the 8192 bytes in the 8 K byte memory module free for normal programming purposes (see figure 4). To determine how many lines your particular television display generator displays, examine the logic which controls the clearing of the line counters and determine the value of the counters just prior to the clearing pulse.

The shift registers convert the parallel data from the programmable memory into a time varying string of video levels in

Figure 3: Schematic Diagram. Taking this "top down" hardware system design one level further for a specific case yields this schematic diagram of the interface as implemented for a Digital Group system and Digital Group television display generator.



Each element of the block diagram in figure 2 is shown here in the form of actual circuitry and wiring. The memory matrix is shown as a generalized box with 8 bank selection inputs for each 1 K bank within the module.



synchronization with the video level information generated by the existing television display generator. Two shift registers are required so that one may be loading data while the other is shifting data out. To insure that the graphics display unit video information is in synchronization with the video of the television display generator, the clock pulse for the television display generator's shift register should be used for the graphics display generator's shift registers. The load or shift input to the graphics display unit shift registers is controlled by the least significant address bit generated by the graphics display's counters. This bit changes state each time that a new byte is accessed in memory. The timing here is critical. The shift register which was loading data must disable its parallel inputs before the change in the address lines affects the data output from the memory. For the 2102-1 integrated circuits this time is a minimum of 50 ns. This shift register must also change from the loading state to the shifting state prior to the next clock pulse from the television display generator. This may require slightly delaying the clock pulse from the TV display generator. In my system, it required approximately 20 ns delay in the clock line, provided by using a pair of inverters in series. A data selector is used to pick the output from the shift register which is currently shifting data.

Figure 3 shows the implementation of this system for a Digital Group 8080A system, using the block diagram of figure 2 to guide the detailed design. In my description which follows, no attempt is made to explain the schematic in detail. Instead, the various components are related to the functional blocks of figure 2.

IC1, IC3 and part of IC5 make up the buffer, block (1) in figure 2, which isolates the processor's address bus from the memory in the display mode. IC2, IC4, and part of IC5 form the buffer, block (2) of figure 2, which isolates the graphics display unit counters from the memory in the update mode. IC6 and IC7 form the buffer, block (3) of figure 2, which isolates the processor's data input bus from the memory in the display mode. (The processor's data output bus is always connected to the input side of the memory, since it is the only source of data written in the memory.) IC13 and IC14 buffer the output from the memory circuits to provide the required output drive current.

IC8 and IC9 are the character counters. Note that the three least significant output bits of IC8 are left unconnected. This in effect divides the output by 8. Since each byte from the memory contains 8 bits of

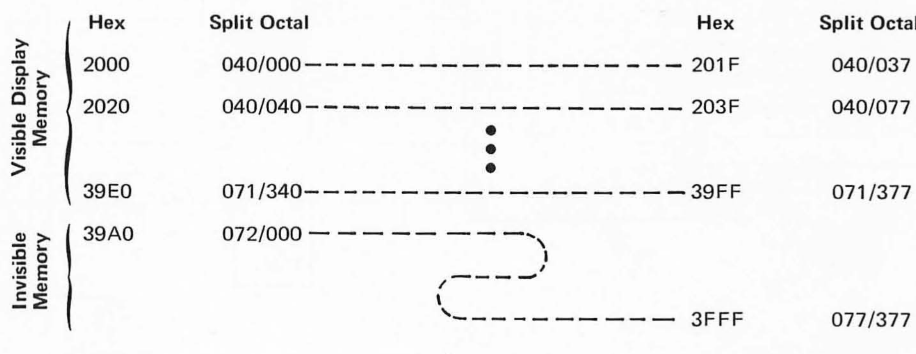


Table 1: Integrated Circuit Power List.

No	Device	+5 V	GND
IC1	8T97	16	8
IC2	8T97	16	8
IC3	8T97	16	8
IC4	8T97	16	8
IC5	8T97	16	8
IC6	8T97	16	8
IC7	8T97	16	8
IC8	74193	16	8
IC9	74193	16	8
IC10	74193	16	8
IC11	74193	16	8
IC12	7442	16	8
IC13	74125	14	7
IC14	74125	14	7
IC15	74165	16	8
IC16	74165	16	8
IC17	74157	16	8
IC18	7404	14	7
IC19	7437	14	7
IC20	7404	14	7
IC21	7410	14	7

Programming Note: Addresses evenly divisible by hexadecimal 20 (decimal 32) in this address space allocation are the beginnings of a 256 bit (32 byte) raster scan line.

Using SHL(I,J) and SHR(I,J) to indicate the 16 bit left and right shift operations on I, by J positions, then given 16 bit integer coordinates X (horizontal coordinate from upper left) and Y (vertical coordinate from upper left down), the byte address offset of the required bit within the displayed memory region is:

AB := SHL(Y, 5) + SHR(X,3) ;

Similarly, the rotation factor required to place a mask bit opposite the desired bit of a picture byte is given by:

ROTFAC := X & 7;

(It is assumed that an 8 bit right rotate is used and the mask starts out with a hexadecimal 80 value.)

Figure 4: Memory address space allocations for the graphics display. This figure charts the specific hexadecimal and split octal address locations for the 208 lines and free area in the author's system. Each line is represented by 32 eight bit bytes. Each byte specifies 8 consecutive dots on the screen, one for each bit. The zone at the end of the map contains the invisible section of the 8 K region devoted to this memory and display subsystem.

information, a new byte is required only after 8 bits are displayed. This assumes that a count pulse is received for each bit displayed. IC10 and IC11 are the line counters. They count the number of scan lines which have been displayed.

IC15 and IC16 are the two shift registers. Note that the clock input to these integrated circuits passes through two inverters in series. This provides the 20 ns delay required to allow the shift registers to change state from the load mode to the shift enabled mode. The output of the shift register which is shifting out data is selected by IC17 to be sent through a NAND gate to the television display generator where it is combined with the existing video display signal. Switch S2, also connected to the NAND gate IC21b, provides the capability to blank the video display from the screen. If this switch is omitted and the wire from IC21 pins 3 and 4 connected to one bit of an output port, programmable blanking of the graphics can be achieved.

IC12 is a BCD to decimal converter which is used to select one of the eight programmable random access memory banks on the 8 K memory card. This integrated circuit converts the contents of address bits A10 to A12 into a bank enable signal which drives the eight 2102s of each 1 K bank within the memory card.

IC18, IC19, IC20 and IC21 provide the mode control logic. The board address selection jumpers are connected to locate the memory at the desired position in address space. The jumpers as shown locate the visible portion of memory in addresses 8192 to 16,383 (hexadecimal 2000 to 3FFF, split octal 040/000 to 077/377). Note that in the case of a television display generator with less than 256 lines visible, the upper portion of the memory range is available for program storage. In the 208 line case of my system, the 1536 invisible bytes are located at hexadecimal addresses 3A00 to 3FFF, split octal addresses 072/000 to 077/377. Figure 4 shows the layout of memory and its relationship to the displayed data. The low addresses in memory define the top of the image; the high addresses in memory define the lower portion of the image.

Since the processor is granted access to the memory only during the vertical blanking pulses in the scheme just described, programs which have critical timing may not work properly. For example, software for a cassette interface has timing requirements asynchronous to the video display and would not work properly if referencing data in the display region of memory. Switch S1 in figure 3 is provided to allow the processor access to memory at any time, without waiting. Thus when loading the display memory from a cassette tape, the conflict between the graphics display wait time and the needs of time critical software is resolved (at the expense of "sparkle" in the display during the loading). For processors which cannot wait for slow memory, S1 should be eliminated and the enable input, IC19 pin 5, should be grounded.

In making the connections from the graphics display to the television display generator and memory board, make all wires as short as possible, certainly not longer than 10 to 11 inches (25 to 28 cm). Depending upon the existing loads on the signals you are tapping, buffers may be required to drive the lines. The power bus should be heavily bypassed with .01μF capacitors from +5 V to ground.

In Conclusion

This graphics display generator can be built very inexpensively, taking advantage of existing circuitry in the form of memory modules and a television display generator. When you've got it up and running, the world of visual imagery will be available to you on your television output, as is demonstrated by the examples seen in photos 1 through 3.■

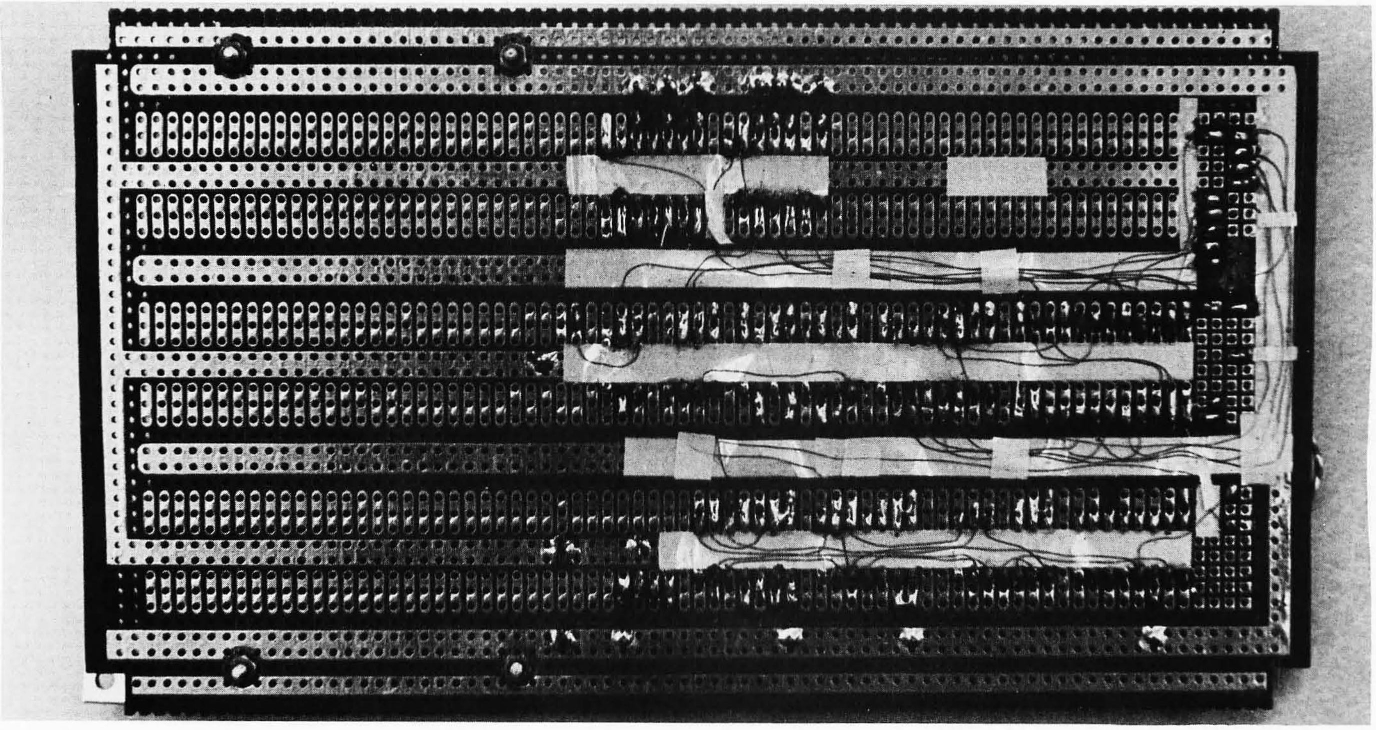


Photo 1.

A Tip for Using Wiring Pencils

R W Burhans
Avionics Engineering Center
Ohio University
Athens OH 45701

One of the problems in using the Vector or other brands of "wiring pencils" is the poor visibility of the fine #36 wire viewed against the typical background of Vector Blue circuit board material. Older types of light green epoxy fiberglass boards give somewhat better contrast and the lower quality epoxy paper board is very easy on the eyes due to its light color. Of the various colors available, the red wire has the best contrast with respect to the newer blue board materials, but even this is sometimes not good enough. The assembly of prototype hardware with this technique can be greatly improved by laying down a layer of yellow insulating tape where the pencil wrap wires are to run. We have used 3M brand yellow coil insulating tape. This is almost exactly like 0.25 inch (0.635 cm) wide tape recorder splicing tape. The yellow tape is placed on the circuit board after the common grounds and positive power buses are hard wired to the socket pins using bus strips or heavy gauge wire. Then the task of trying to see the fine wrap wires becomes much easier for the remainder of the assembly. An example showing the underside wiring of an Omega-VLF receiver interface for use with a JOLT microprocessor is shown in photo 1. [*More about the Omega microprocessor interfaces will appear in a future issue of BYTE.*] Small strips of the yellow insulating tape are cut and pasted to help hold the wires in place for semipermanent prototype circuits. In general we like to use hard soldered connections for prototype work where the microprocessor interface hardware is used in mobile, marine, or airborne equipment. In such applications, the effects of vibration are a major reliability consideration, hence the need of solder as a "glue" for the wiring. A combination of top of board jumper wiring and the Vector wire wrap and solder technique results in less eye strain during assembly when the yellow insulating tape is used to improve the wire visibility.■

SYNCHRO-SOUND ENTERPRISES

PRESENTS

ADM-3 KIT | LSI-11



24 line X 80 column
VIDEO TERMINAL KIT

16 BIT MICROPROCESSOR
400 MACHINE INSTRUCTIONS
with 4K X 16 RAM

WINTER SPECIALS ADM-3K $849.95 LSI-11 $840.00 IMSAI 8080 $559.95

FEATURING THE FOLLOWING PRODUCTS

LEAR SIEGLER
IMS ASSOCIATES, INC.
CROMEMCO
SOUTHWEST TECHNICAL PRODUCTS
TECHNICAL DESIGN LABS (Z-80)
DIGITAL EQUIPMENT CORPORATION
DEC WRITERS
PROCESSOR TECHNOLOGY
"SEALS" RAM BOARDS
OLIVER AUDIO ENGINEERING

WORKING MODELS ON DISPLAY AT OUR NEW SHOWROOM

CALL OR WRITE FOR
DISCOUNT PRICE LIST
PHONE ORDERS
(212) 468-7067

STORE HOURS SAT (9AM—4PM)
WEEKDAYS CALL
SYNCHRO-SOUND ENTERPRISES
193-25 JAMAICA AV
HOLLIS NY 11423

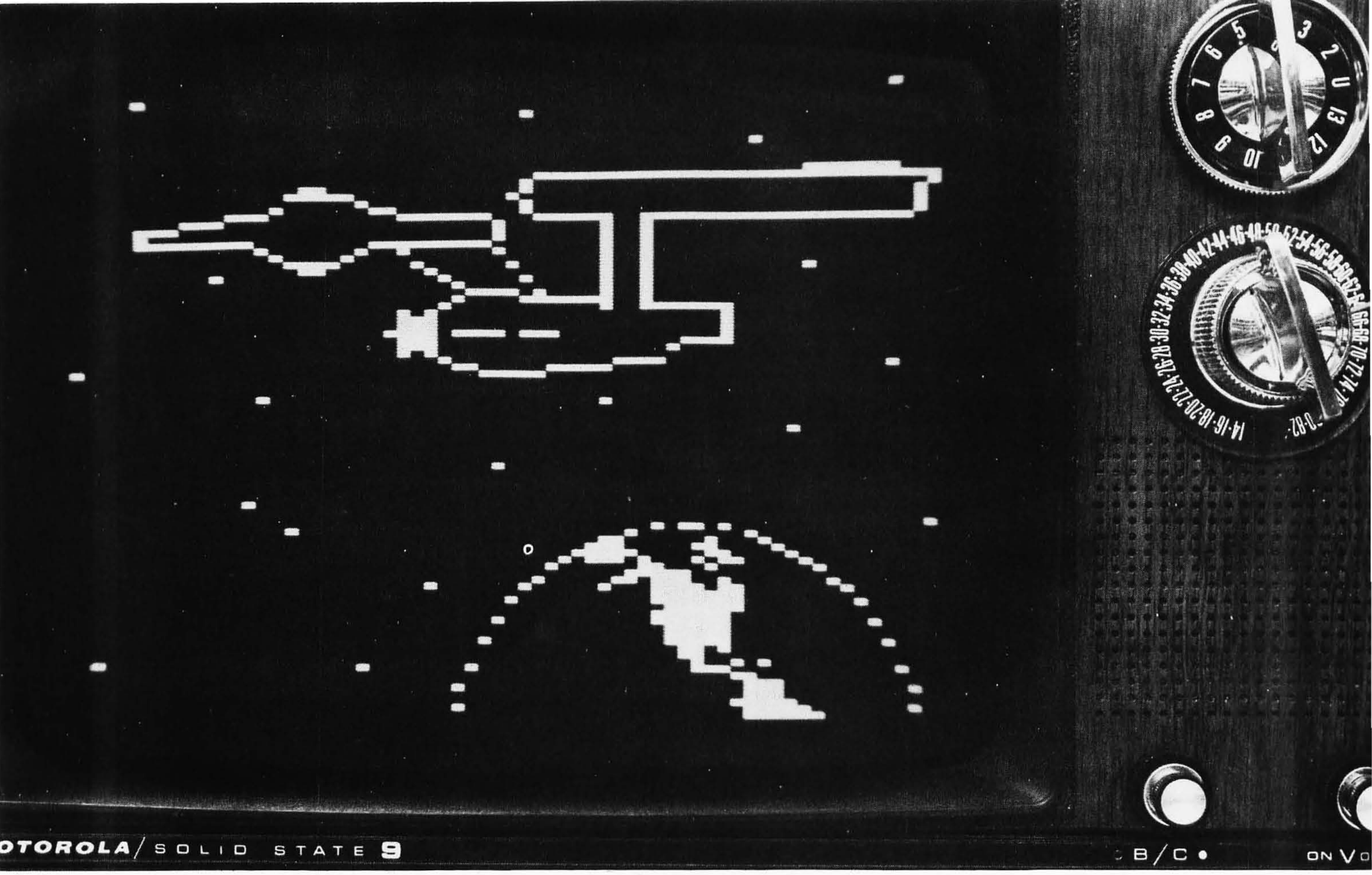


Photo 1: Using the data of listing 2 with the program of listing 1, this picture of a famous space craft can be placed on a TV screen using the Southwest Technical Products GT-6144 display. The actual picture is created from three data table segments which define the Enterprise, the Earth and the background of "stars."

An Enterprising Display Device

Joe Deres
Southwest Technical Products Corp
219 W Rhapsody
San Antonio TX 78216

Many people find games one of the most interesting applications of a computer system. These range from simple games such as Tic Tac Toe, Blackjack and Ping Pong to very complicated games such as Space War. All of these have one thing in common; they are much more fun when played using a video graphics display. With such terminals you can provide instant response and provide a pictorial playing area that cannot be duplicated on any type printing terminal. Can you imagine attempting to play an elaborate game on a Teletype machine? Unfortunately, the typical high resolution graphics display normally costs thousands of dollars and is completely beyond the budget of the average person.

The GT-6144 is our answer to this dilemma. By settling for a bit less resolution than is available in expensive graphics terminals we can generate graphic displays on any monitor (or a standard TV set to which a video input has been added).

In this design, the display screen is divided into an array of cells 64 positions wide by 96 positions high. Each cell is individually addressable and may be selectively turned on or off by programmed commands from the parallel interface to your computer. With a little imaginative programming (and lots of memory) fixed or

moving images may be displayed on the screen for added enhancement to game programs. Photo 1 shows Star Trek's star ship, the "USS Enterprise," generated with the GT-6144 by running the program of listing 1 using data of listing 2 in an SWTPC 6800 Computer System. It takes only 2 μs to load a single cell of the display memory; much faster than most contemporary personal computers can generate the information. The system features a power up screen blanking circuit which in addition may be enabled, or disabled at any time through program commands from the computer system or hardwired switches. In addition, an image reversing feature allows you to select between white on black or black on white images by a simple one word command generated by your computer's program. The system will operate on either 50 Hz or 60 Hz power lines with American standard 525 line or European standard 625 line television sets or video monitors.

The GT-6144 design incorporates a 6,144 bit static programmable memory which eliminates the requirement that it be used with a specific computer system. The terminal will operate with any computer system whose parallel TTL level interface outputs an 8 bit data word and data ready strobe. The hookup problem then reduces to making sure that each bit line at the interface runs to the corresponding bit at the input to the GT-6144.

The unit is available as a kit which is complete less the chassis and does not include the required video monitor or modified television set. Instructions for the addition of a video input jack to a typical television are included with the kit, and a switch installed on the back of the TV set will allow one to select between terminal and normal television operation. You may use the same television set or video monitor used by the CT-1024 terminal system. In fact, control commands from your computer allow you to display graphic, CT-1024 alphanumeric, or even a combination of the two, all on the same display device. The mixing of graphic and alphanumeric video is engineered to work with an SWTPC CT-1024 terminal. The video from other alpha-

Listing 1: A GT-6144 Demonstration Program. This Motorola 6800 program, written to illustrate the operation of the GT-6144, contains routines to erase the screen, to write a pattern using a table of commands, and to select various options under control of a simple interpreter driven by the terminal keyboard. The routine is shown in assembly language format, with an origin at location 0000. This program must be run in programmable memory without write protection since several data items are located within the program text. Relocation of the program at arbitrary addresses other than its present page 0 origin will in general require modifications of symbolic code and reassembly to eliminate direct addressing at several points. There are several gaps in the addressing sequence used, reflecting the fact that the program was hand assembled. The data used to generate the picture seen in photo 1 is summarized in listing 2.

```
Address  Hex Code    Label      Op     Operand    Commentary

                     * This program, by Joe Deres, is designed to transfer
                     * a data table of GT-6144 commands to the display.
                     * The main entry point is at location ENTER, address 0016
0000     00 00       START      RMB    2          Pointer to data;
0002     00 00                  RMB    2
0004     80 0C       PIA3       RMB    2          Pointer to PIA address of GT-6144;
         00 05       PIA3L      EQU    PIA3+1

                     * The following subroutine sends the contents of accumulator A
                     *    to the GT-6144 and generates the active low "data ready"
                     *    strobe at the peripheral control pin of the PIA.
0006     FE 00 04    OUTGR      LDX    PIA3
0009     A7 00                  STAA   0,X        PIA3[data] := A [send data to GT-6144];
000B     C6 37                  LDAB   #$37
000D     E7 01                  STAB   1,X        PIA3[control] := code for low strobe;
000F     E6 00                  LDAB   0,X        B := PIA3[data] [dummy op for timing];
0011     C6 3F                  LDAB   #$3F
0013     E7 01                  STAB   1,X        PIA3[control] := code for high strobe;
0015     39                     RTS

                     * The following is the initialization sequence for the program,
                     *    and the subroutine calls which invoke the picture generation;
                     *    The program is entered by calling ENTER with a JSR;
0016     86 3C       ENTER      LDAA   #$3C
0018     B7 80 07               STAA   $8007      Disable control interface echo option;
001B     FE 00 04               LDX    PIA3       X := PIA3 address pointer;
001E     C6 FF                  LDAB   #$FF
0020     E7 00                  STAB   0,X        PIA3[direction] := code for output all bits;
0022     C6 3F                  LDAB   #$3F
0024     E7 01                  STAB   1,X        PIA3[control] := code for high strobe,
                     *                               and point to data register;
0026     BD 00 A2               JSR    ERASE      clear the screen with subroutine;
0029     7E 00 D0               JMP    SHOW       fill screen with USS Enterprise or other
                     *                               suitable data table;

                     * The following subroutine, with entry at PUTGRPHX, is used to run
                     *    through a data table stored beginning at the address in START;
0030     FE 00 00    NXTDATA    LDX    START
0033     08                     INX               START := START + 1 [16 bits precision];
0034     FF 00 00               STX    START
0037     FE 00 00    PUTGRPHX   LDX    START
003A     A6 00                  LDAA   0,X        if @START = FF [leave when data = FF];
003C     81 FF                  CMPA   #$FF
003E     27 16                  BEQ    ENDPUT     then go to ENDPUT;
0040     8B 00       DISP1      ADDA   #0         caution: program modifies immediate byte!
0042     BD 00 06    NEWXSPOT   JSR    OUTGR
0045     FE 00 00               LDX    START
0048     08                     INX               START := START + 1;
0049     FF 00 00               STX    START
004C     A6 00                  LDAA   0,X
004E     81 FE                  CMPA   #$FE       if @START = FE then go to NXTDATA
0050     27 DE                  BEQ    NXTDATA       [treat as horizontal position if FE];
0052     8B 00       DISP2      ADDA   #0         caution: program modifies immediate byte!
0054     20 EC                  BRA    NEWXSPOT
0056     39          ENDPUT     RTS               return to caller;

                     * The following DISPLAY routine is used to set up the PUTGRPHX
                     *    routine, by modifying two bytes of code and setting up the
                     *    initial pointer value in START at location 0000;
0057     8B 40       DISPLAY    ADDA   #$40       [A is defined prior to entry];
0059     B7 00 41               STAA   DISP1+1    (DISP1+1) := #$40 + A;
005C     17                     TBA               [B is defined prior to entry];
005D     8B 80                  ADDA   #$80
005F     B7 00 53               STAA   DISP2+1    (DISP2+1) := #$80 + B;
0062     FF 00 00               STX    START      START := X [X is defined prior to entry];
0065     BD 00 37               JSR    PUTGRPHX   call graphics put routine;
0068     39                     RTS               return to caller;

                     * DELAY is a routine to generate a delay in processing,
                     *    the parameter (approximately calibrated in seconds) is passed
                     *    in the B accumulator; routine uses A and X;
0070     86 00       DELAY      LDAA   #0         A := 0 [outer loop count];
0072     CE 00 00    NEWINNER   LDX    #0         X := 0 [inner loop count (time calibration)];
0075     08          DELLOOP    INX               X := X + 1;
0076     8C FF FF               CPX    #$FFFF     if X = FFFF
0079     27 02                  BEQ    OUTER      then go to OUTER loop check;
007B     20 F8                  BRA    DELLOOP    else reiterate inner loop;
007D     4C          OUTER      INCA              A := A + 1;
007E     11                     CBA               if A = B
007F     27 02                  BEQ    DELAYDON   then go to DELAYDON;
0081     20 EF                  BRA    NEWINNER   else go to NEWINNER [to start new inner loop];
0083     39          DELAYDON   RTS               return to caller;

                     * The ERASE subroutine in locations 00A0 to 00CC is
                     *    used to clear the GT-6144 screen prior to establishing
                     *    a new picture; it does this by addressing each
                     *    display location in turn and setting a null state;
```

Listing 1, continued:

```
Address Hex Code    Label     Op     Operand   Commentary

00A0    00          HPOS      RMB    1         horizontal position data byte;
00A1    00          VPOS      RMB    1         vertical position data byte;
00A2    96 A0       ERASE     LDAA   HPOS      [entry point, outer loop for horizontal position];
00A4    81 40                 CMPA   #$40      if HPOS = $40 [is it end of outer loop]?
00A6    27 1E                 BEQ    OUT       then go to OUT [erasure is done];
00A8    01 01                 NOP              [two do nothing NOPs];
00AA    BD 00 06              JSR    OUTGR     define horizontal coordinate;
00AD    96 A1       ERASEV    LDAA   VPOS      [inner loop for vertical position];
00AF    81 60                 CMPA   #$60      if VPOS = $60 [is it end of inner loop]?
00B1    27 0A                 BEQ    SPEC      then go to SPEC [and restart inner loop];
00B3    8B 80                 ADDA   #$80      convert coordinate to vertical command;
00B5    BD 00 06              JSR    OUTGR     define vertical coordinate and value;
00B8    7C 00 A1              INC    VPOS      VPOS := VPOS + 1;
00BB    20 F0                 BRA    ERASEV    reiterate inner loop;
00BD    86 00       SPEC      LDAA   #$00
00BF    97 A1                 STAA   $A1       VPOS := 0;
00C1    7C 00 A0              INC    HPOS      HPOS := HPOS + 1;
00C4    20 DC                 BRA    ERASE     reiterate outer loop;
00C6    86 00       OUT       LDAA   #$00
00C8    97 A0                 STAA   HPOS      HPOS := 0;
00CA    97 A1                 STAA   VPOS      VPOS := 0;
00CC    39                    RTS              return to caller;

                    * The SHOW routine is used to define the parameters of DISPLAY,
                    *   pointing to the Enterprise screen location and its data table,
                    *   then to call DISPLAY; since the vertical and horizontal
                    *   positions are parameters, the picture can be located at arbitrary
                    *   places within the display area of the GT-6144;
00D0    86 05       SHOW      LDAA   #$05      locate Enterprise 5 cells across,
00D2    C6 0D                 LDAB   #$0D        and 13 cells down;
00D4    CE 02 00              LDX    #$200     point to Enterprise pattern data;
00D7    BD 00 57              JSR    DISPLAY   call DISPLAY to draw the Enterprise;
00DA    86 1B                 LDAA   #$1B      locate planet Earth 27 cells across,
00DC    C6 40                 LDAB   #$40        and 64 cells down;
00DE    CE 04 00              LDX    #$400     point to Earth pattern data;
00E1    BD 00 57              JSR    DISPLAY   call DISPLAY to flash the Earth;
00E4    C6 0A                 LDAB   #$0A      set approximately 10 second delay parameter;
00E6    BD 00 70              JSR    DELAY
00E9    86 00                 LDAA   #$00      locate random "star" data 0 cells across,
00EB    C6 00                 LDAB   #$00        and 0 cells down;
00ED    CE 05 00              LDX    #$500     point to star pattern data;
00F0    BD 00 57              JSR    DISPLAY   call DISPLAY to draw in the stars;
00F3    7E 01 00              JMP    OPTFUNC   go to OPTFUNC [to perform optional functions];

                    * The OPTFUNC routine is reached after the display is set up
                    *   on the screen and is used to interpret several commands
                    *   from the terminal keyboard in order to toggle some of the
                    *   options of the display unit;
0100    BD E1 AC    OPTFUNC   JSR    INEEE     call INEEE (MIKBUG subroutine) to get
                    *                            a single character of input in A;
                    * What follows is a simple series of tests to identify one of
                    * eight commands selected by ASCII characters '1' through '7';
0103    81 31                 CMPA   '1'       if A = '1'
0105    27 1E                 BEQ    ONE       then go to ONE;
0107    81 32                 CMPA   '2'       if A = '2'
0109    27 1E                 BEQ    TWO       then go to TWO;
010B    81 33                 CMPA   '3'       if A = '3'
010D    27 1E                 BEQ    THREE     then go to THREE;
010F    81 34                 CMPA   '4'       if A = '4'
0111    27 1E                 BEQ    FOUR      then go to FOUR;
0113    81 35                 CMPA   '5'       if A = '5'
0115    27 1E                 BEQ    FIVE      then go to FIVE;
0117    81 36                 CMPA   '6'       if A = '6'
0119    27 1E                 BEQ    SIX       then go to SIX;
011B    81 37                 CMPA   '7'       if A = '7'
011D    27 1E                 BEQ    SEVEN     then go to SEVEN;
011F    81 38                 CMPA   '8'       if A = '8'
0121    27 1E                 BEQ    EIGHT     then go to EIGHT;
0123    20 DB                 BRA    OPTFUNC   try again, over and over forever;

                    * Each individual command is implemented by defining a constant
                    * in the A accumulator, then jumping to EXECUTE in order to call
                    * OUTGR and modify the state of the GT-6144;
0125    86 E0       ONE       LDAA   #$E0      ONE:     Set normal screen mode;
0127    20 1A                 BRA    EXECUTE
0129    86 E1       TWO       LDAA   #$E1      TWO:     Enable CT-1024; display;
012B    20 16                 BRA    EXECUTE
012D    86 E2       THREE     LDAA   #$E2      THREE:   Disable graphics display;
012F    20 12                 BRA    EXECUTE
0131    86 E3       FOUR      LDAA   #$E3      FOUR:    Unused bit pattern;
0133    20 0E                 BRA    EXECUTE
0135    86 E4       FIVE      LDAA   #$E4      FIVE:    Set inverted screen mode;
0137    20 0A                 BRA    EXECUTE
0139    86 E5       SIX       LDAA   #$E5      SIX:     Disable CT-1024; display;
013B    20 06                 BRA    EXECUTE
013D    86 E6       SEVEN     LDAA   #$E6      SEVEN:   Enable graphics;
013F    20 02                 BRA    EXECUTE
0141    86 E7       EIGHT     LDAA   #$E7      EIGHT:   Unused bit pattern;
0143    BD 00 06    EXECUTE   JSR    OUTGR     call OUTGR to select option;
0146    20 B8                 BRA    OPTFUNC   reiterate forever;
```

```
0000    START       Graphics table pointer;
0004    PIA3        8 bit output PIA port;
0006    OUTGR       Graphics output routine;
0016    ENTER       Main entry of graphics demo program;
0030    NXTDATA
0037    PUTGRPHX    Graphics put routine;
0040    DISP1
0042    NEWXSPOT
0052    DISP2
0056    ENDPUT
0057    DISPLAY     Graphics put routine setup;
0070    DELAY       Delay routine (gross calibration in seconds);
0072    NEWINNER
0075    DELLOOP
007D    OUTER
0083    DELAYDON
00A0    HPOS        ERASE horizontal position;
00A1    VPOS        ERASE vertical position;
00A2    ERASE       Subroutine to clear the screen;
00AD    ERASEV
00BD    SPEC
00C6    OUT
00D0    SHOW        Routine to sequence display of photo 1;
0100    OPTFUNC     Options selection interpreter (keyboard driven);
0125    ONE
0129    TWO
012D    THREE
0131    FOUR
0135    FIVE
0139    SIX
013D    SEVEN
0141    EIGHT
0143    EXECUTE
```

numeric terminals will not necessarily work with the GT-6144 since the unit was specifically designed to work with the sync signals of the CT-1024. Power requirements for the terminal are 5.0 VDC @ 1 A, −12 VDC @ 20 mA and 6 VAC @ 20 mA.

Programming

The display of the GT-6144 graphics display generator produces a matrix of 6144 small rectangular cells formatted 64 across and 96 down. Each of these cells can be turned on or off at will. In order for the GT-6144 to do a particular function the data fed to it must be formatted correctly. Figure 1 summarizes the formatting conventions. The coordinate of a particular location is referenced from the top left corner of the screen with the first square residing at location (0,0). When sending data to the GT-6144, the first parallel byte sent to the terminal must be the horizontal position coordinate. The actual position is determined by the binary number in low order bits B5 to B0. When bit B6 is 0, a rectangle will be removed at the desired coordinates; when B6 is 1, a white rectangle will be generated. Bit 7 must always be 0 for the display to know that a horizontal position is being loaded. A 0 in bit B7 causes the data holding flip flops in the terminal to store the present data.

The second byte from the computer contains the vertical coordinate. The location is contained in binary in bits B6 to B0 of this second byte while bit 7 must equal a 1.

When programming an image to appear on the screen there are two ways the characters can be loaded. The method used depends on how your software is organized. One method is to just send out successive coordinates (H_1, V_1), (H_2, V_2) etc, until all H, V locations are specified. With this method two bytes must be sent out for each character.

Another method can be used that will result in saving time and memory space. In this method the horizontal position of a particular column is loaded only once into

the terminal. The vertical coordinate of all other characters that have this same horizontal coordinate can then be loaded by themselves since the horizontal position is latched in the terminal's holding flip flops. This second method is used by the programs accompanying this article.

Since there are 96 characters to be accessed in the vertical direction at least seven address lines must be used. Seven lines give the possibility of addressing 2^7 (128 decimal) locations giving us 32 extra undefined locations. These extras can be used as control commands for controlling blanking on and off, reverse screen, etc. The format for control commands for the GT-6144 display generator is found in figure 2.

When writing programs care should be taken to optimize them for speed and memory conservation. All of the above functions can also be under hardware control by using SPST push button switches.

USS Enterprise data 0200 → 0364

0200 00	022D 0B	025A 13	0287 08	02B4 FE	02E1 24	030D 08	0339 01
0201 0A	022E 04	025B 08	0288 0C	02B5 1E	02E2 02	030E 15	033A 02
0202 0B	022F 05	025C 0C	0289 13	02B6 02	02E3 08	030F 1A	033B 08
0203 0C	0230 0F	025D 0F	028A 19	02B7 08	02E4 09	0310 FE	033C FE
0204 FE	0231 10	025E 17	028B 1F	02B8 14	02E5 0A	0311 2A	033D 33
0205 01	0232 FE	025F 18	028C FE	02B9 1E	02E6 0B	0312 02	033E 01
0206 0A	0233 0C	0260 19	028D 19	02BA FE	02E7 0C	0313 08	033F 02
0207 0C	0234 04	0261 1A	028E 09	02BB 1F	02E8 0D	0314 16	0340 08
0208 FE	0235 05	0262 1B	028F 0A	02BC 02	02E9 0E	0315 17	0341 FE
0209 02	0236 0F	0263 FE	0290 0B	02BD 08	02EA 0F	0316 18	0342 34
020A 0A	0237 10	0264 14	0291 0D	02BE 14	02EB 10	0317 19	0343 01
020B 0C	0238 FE	0265 08	0292 13	02BF 1E	02EC 11	0318 1A	0344 02
020C FE	0239 0D	0266 0C	0293 19	02C0 FE	02ED 12	0319 FE	0345 08
020D 03	023A 05	0267 10	0294 1F	02C1 20	02EE 13	031A 2B	0346 FE
020E 09	023B 0F	0268 16	0295 FE	02C2 02	02EF 14	031B 02	0347 35
020F 0C	023C FE	0269 17	0296 1A	02C3 08	02F0 15	031C 08	0348 01
0210 FE	023D 0E	026A 18	0297 05	02C4 14	02F1 1D	031D FE	0349 02
0211 04	023E 05	026B 19	0298 0F	02C5 1E	02F2 FE	031E 2C	034A 08
0212 09	023F 0E	026C 1A	0299 13	02C6 FE	02F3 25	031F 02	034B 08
0213 0C	0240 FE	026D 1B	029A 1F	02C7 21	02F4 02	0320 08	034C FE
0214 FE	0241 0F	026E 1C	029B FE	02C8 02	02F5 08	0321 FE	034D 36
0215 05	0242 06	026F FE	029C 1B	02C9 08	02F6 15	0322 2D	034E 01
0216 09	0243 0D	0270 15	029D 03	02CA 09	02F7 1D	0323 02	034F 02
0217 0C	0244 FE	0271 08	029E 04	02CB 0A	02F8 FE	0324 08	0350 08
0218 FE	0245 10	0272 0C	029F 06	02CC 0B	02F9 26	0325 FE	0351 FE
0219 06	0246 07	0273 11	02A0 07	02CD 0C	02FA 02	0326 2E	0352 37
021A 08	0247 0C	0274 15	02A1 11	02CE 0D	02FB 08	0327 02	0353 02
021B 0C	0248 FE	0275 1D	02A2 14	02CF 0E	02FC 15	0328 08	0354 08
021C FE	0249 11	0276 FE	02A3 19	02D0 0F	02FD 1B	0329 FE	0355 FE
021D 07	024A 07	0277 16	02A4 1F	02D1 10	02FE FE	032A 2F	0356 38
021E 08	024B 0C	0278 08	02A5 FE	02D2 11	02FF 27	032B 02	0357 02
021F 0C	024C 19	0279 0C	02A6 1C	02D3 12	0300 02	032C 08	0358 04
0220 FE	024D FE	027A 12	02A7 02	02D4 13	0301 08	032D FE	0359 05
0221 08	024E 12	027B 14	02A8 08	02D5 14	0302 15	032E 30	035A 06
0222 08	024F 08	027C 19	02A9 13	02D6 15	0303 1A	032F 01	035B 07
0223 0D	0250 0C	027D 1E	02AA 14	02D7 1E	0304 FE	0330 02	035C FE
0224 FE	0251 0D	027E FE	02AB 19	02D8 FE	0305 28	0331 08	035D 39
0225 09	0252 16	027F 17	02AC 1F	02D9 22	0306 02	0332 FE	035E 02
0226 07	0253 17	0280 08	02AD FE	02DA 02	0307 08	0333 31	035F 03
0227 0E	0254 18	0281 0C	02AE 1D	02DB 1D	0308 15	0334 01	0360 FE
0228 FE	0255 19	0282 13	02AF 02	02DC FE	0309 1A	0335 02	0361 2A
0229 0A	0256 1A	0283 19	02B0 08	02DD 23	030A FE	0336 08	0362 15
022A 05	0257 1B	0284 1E	02B1 14	02DE 02	030B 29	0337 FE	0363 FE
022B 0F	0258 1C	0285 FE	02B2 19	02DF 1D	030C 02	0338 32	0364 FF
022C FE	0259 FE	0286 18	02B3 1E	02E0 FE			

Planet data 0400 → 04EF

0400 00	041E 06	043C 09	045A 0B	0478 13	0496 17	04B4 1C	04D2 1F
0401 1A	041F FE	043D FE	045B 0C	0479 04	0497 FE	04B5 1D	04D3 FE
0402 1D	0420 0A	043E 0F	045C 0D	047A 05	0498 15	04B6 1E	04D4 1B
0403 1F	0421 05	043F 02	045D 0E	047B 06	0499 07	04B7 1F	04D5 08
0404 FE	0422 06	0440 08	045E 0F	047C 08	049A 0B	04B8 1F	04D6 1E
0405 01	0423 07	0441 09	045F 10	047D 0C	049B 0C	04B9 FE	04D7 FE
0406 17	0424 FE	0442 0A	0460 11	047E 0D	049C 0D	04BA 18	04D8 1C
0407 FE	0425 0B	0443 0B	0461 12	047F 0E	049D 0E	04BB 05	04D9 0A
0408 02	0426 04	0444 0C	0462 13	0480 0F	049E 16	04BC 19	04DA FE
0409 13	0427 05	0445 0D	0463 14	0481 10	049F 18	04BD 1A	04DB 1D
040A FE	0428 06	0446 0F	0464 15	0482 11	04A0 1C	04BE 1B	04DC 0B
040B 03	0429 07	0447 10	0465 FE	0483 12	04A1 FE	04BF 1C	04DD FE
040C 10	042A 0B	0448 FE	0466 12	0484 13	04A2 16	04C0 1D	04DE 1E
040D FE	042B FE	0449 10	0467 02	0485 17	04A3 03	04C1 1E	04DF 0D
040E 04	042C 0C	044A 09	0468 05	0486 FE	04A4 18	04C2 1F	04E0 FE
040F 0D	042D 04	044B 0A	0469 07	0487 14	04A5 19	04C3 1F	04E1 1F
0410 FE	042E 05	044C 0B	046A 0B	0488 02	04A6 1A	04C4 FE	04E2 10
0411 05	042F 06	044D 0C	046B 0C	0489 06	04A7 1B	04C5 19	04E3 FE
0412 0B	0430 07	044E 0D	046C 0D	048A 07	04A8 1C	04C6 06	04E4 20
0413 FE	0431 0A	044F 0E	046D 0E	048B 0A	04A9 1D	04C7 1C	04E5 13
0414 06	0432 FE	0450 0F	046E 0F	048C 0B	04AA 1E	04C8 1D	04E6 FE
0415 0A	0433 0D	0451 10	046F 10	048D 0C	04AB 1F	04C9 1E	04E7 21
0416 FE	0434 03	0452 11	0470 11	048E 0D	04AC 1F	04CA 1F	04E8 17
0417 07	0435 06	0453 12	0471 12	048F 0E	04AD FE	04CB 1F	04E9 FE
0418 08	0436 09	0454 13	0472 13	0490 0F	04AE 17	04CC FE	04EA 22
0419 FE	0437 0A	0455 FE	0473 14	0491 10	04AF 04	04CD 1A	04EB 1A
041A 08	0438 FE	0456 11	0474 15	0492 11	04B0 16	04CE 07	04EC 1D
041B 07	0439 0E	0457 02	0475 16	0493 12	04B1 19	04CF 1D	04ED 1F
041C FE	043A 07	0458 09	0476 17	0494 13	04B2 1A	04D0 1E	04EE FE
041D 09	043B 08	0459 0A	0477 FE	0495 14	04B3 1B	04D1 1F	04EF FF

Star data 0500 → 0534

0500 01	0507 04	050E FE	0515 14	051C 3A	0523 FE	0529 FE	025F 2A
0501 2C	0508 FE	050F 0D	0516 58	051D FE	0524 31	052A 35	0530 FE
0502 FE	0509 0A	0510 30	0517 FE	051E 20	0525 0A	052B 1C	0531 3E
0503 02	050A 1E	0511 FE	0518 19	051F 06	0526 FE	052C FE	0532 42
0504 58	050B FE	0512 0F	0519 4C	0520 FE	0527 34	052D 3B	0533 FE
0505 FE	050C 0C	0513 44	051A FE	0521 26	0528 34	052E 02	0534 FF
0506 05	050D 40	0514 FE	051B 1E	0522 30			

Listing 2: It takes data to produce a picture. Here is a listing of the data which is used by the demonstration program of listing 1 to produce the picture of the USS Enterprise.

Notes on Construction of a GT-6144

This article contains the complete schematic and enough details for an individual to wire up a version of the display using parts obtained from surplus suppliers and hand wiring techniques such as wirewrap or point to point soldering.

The GT-6144 graphics display generator is also available from Southwest Technical Products Corp, 219 W Rhapsody, San Antonio TX 78216, as a complete kit of parts including the circuit board. The price of this kit is $98.50 postpaid in the United States.

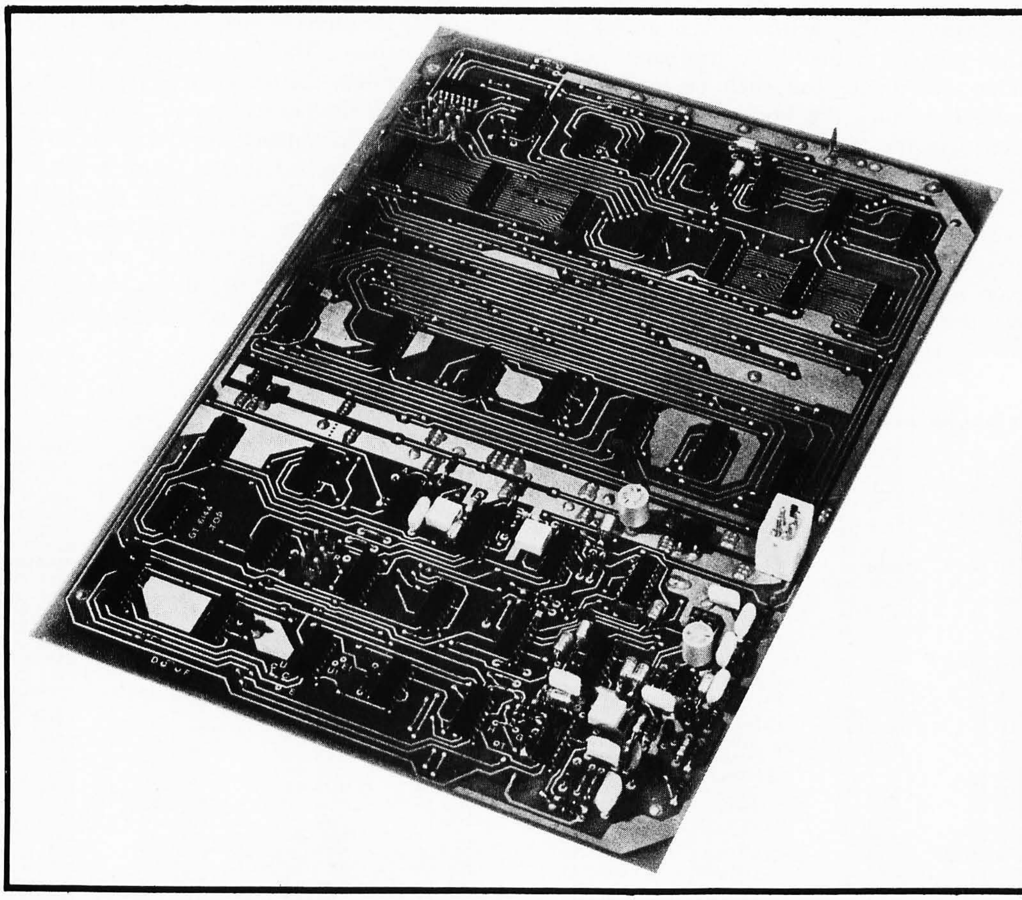


Photo 2: Here is a picture of the GT-6144 as constructed using the printed circuit board and parts supplied with the Southwest Technical Products kit. The logic diagrams of figures 1 to 3 are complete and based upon the original blueprint supplied by the author. Home brewers can use the design information of this article to wire up the circuit described here, or as a starting point in designing a customized version.

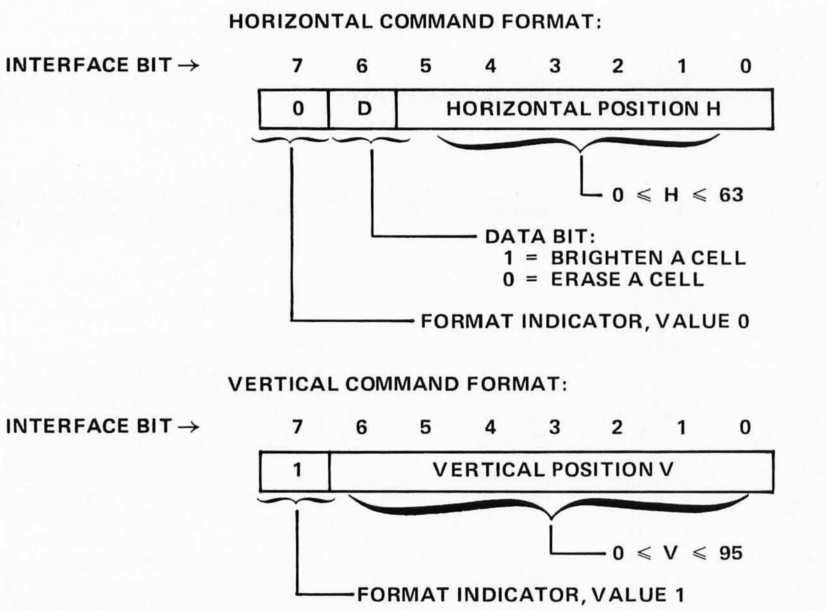


Figure 1: Data Formats for the Parallel Interface to the GT-6144. The GT-6144 has two data formats. The horizontal command format is used to set the sense of the data to be written and load a 6 bit integer horizontal coordinate. Once the horizontal coordinate has been set up, the vertical command format is used to pick a vertical line with a 7 bit integer less than decimal 95 in value, and write the data value ("on" or "off") selected with the horizontal command.

How It Works

The entire screen of this video display has been arranged for 96 lines of 64 characters per line. In addition to the 96 lines of video data the television or monitor also requires the terminal to generate vertical and horizontal sync pulses. The GT-6144 consists of two major sections. The timing sections (figure 3) generate the necessary sync pulses, while the memory section (figures 4 and 5) takes care of storing the necessary information and recalling it while a frame is being generated.

The timebase oscillator as indicated in the schematic is responsible for initiating the horizontal sync pulse and for starting the chain of events that will generate one line of video data to be displayed. The circuit itself is a phase locked loop employed as a frequency multiplier. A 555, IC1, is used as an astable voltage controlled oscillator. The 2N5139 transistor Q3 and TIS58 field effect transistor Q1 along with capacitor C5 form a sample and hold circuit which feeds IC1's voltage control input through the TIS58 field effect transistor Q2. The sample and hold in this case is being used as a phase comparator providing an output voltage proportional to the phase difference of the 60 Hz power line and the multiplied output frequency of the 555 oscillator IC1. The

actual amount of frequency multiplication is equal to the amount of frequency division between the output of the oscillator IC1 and the 60 Hz input reference frequency. As you will see later the value of the frequency divider is 264 and since our reference is 60 Hz so that:

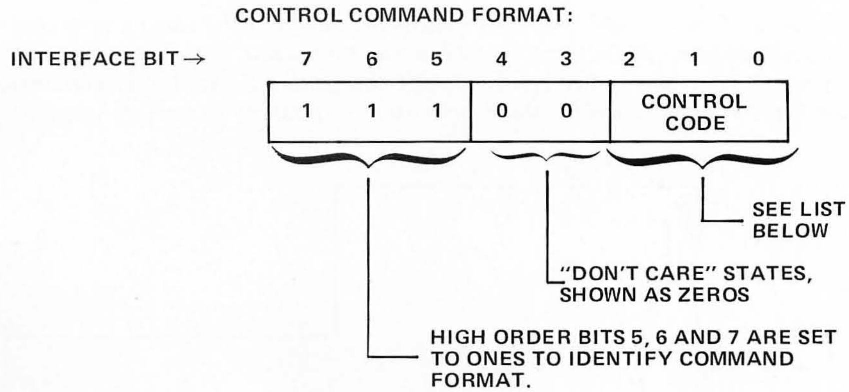
$$fo = (60\ Hz)*(264) = 15{,}840\ Hz$$

which is very close to the horizontal oscillator frequency of a standard television set.

The output of IC1 is fed via inverter IC23c across a jumper to IC17 sections b and c, where among other things a 4 μs horizontal sync pulse is generated. From here the pulse is routed to IC15b where it is ORed with the vertical sync pulse which will be described in detail later.

The falling edge of this sync pulse at the output of IC17c triggers IC16a, a one shot which puts out a positive pulse on pin 4 adjustable by potentiometer R20 from 10 to 30 μs in length. The delay pulse creates a lag between the start of a line and the beginning of data, thus giving an adjustable left margin. Pin 4 of IC16a inhibits dot oscillator IC16b through AND-OR-INVERT gate of IC10a. Pin 13 of IC16a resets 7493 counters IC19 and IC 13, the 16 bit counters which address one of the horizontal positions on a line. Since we are just starting a new line we must first clear the counter to prepare it for incoming data. At the end of a high to low transition of pin 4 of IC16, the 7493 vertical position counter IC6 is incremented; and if there is a ripple carry, IC7 is incremented as well. IC6 and IC7 together keep track of the horizontal scan line presently in use. Together these two counters provide a unique binary code for each of the 96 * 2 = 192 video scan lines generated by the GT-6144.

Now for those of you who are familar with television circuits, you probably know that we need more like 264 lines and not 192 for a complete frame. When a count of 192 (the 193rd line) is reached, IC15a sends out a clock pulse to IC4a and to IC9c which resets the line counters. When IC4a is clocked pin 6 goes low, giving a blanking pulse to prevent any video data from being transmitted from the GT-6144. When pin 6 of IC4 is low, pin 5 is high and enables decoders IC11a and IC11b. When the line counters reach a count of 32, IC11a triggers and remains so for 8 lines until count 40 is reached. The low output on pin 6 during this time is the vertical sync pulse, which goes to IC15b as mentioned earlier. At count 72 IC11b triggers clearing flip flop IC4a which unblanks the video and resets line counters IC6 and IC7 for the generation of another line of displayed data. The above circuitry gives us a total of 192 + 72 = 264 scan lines.

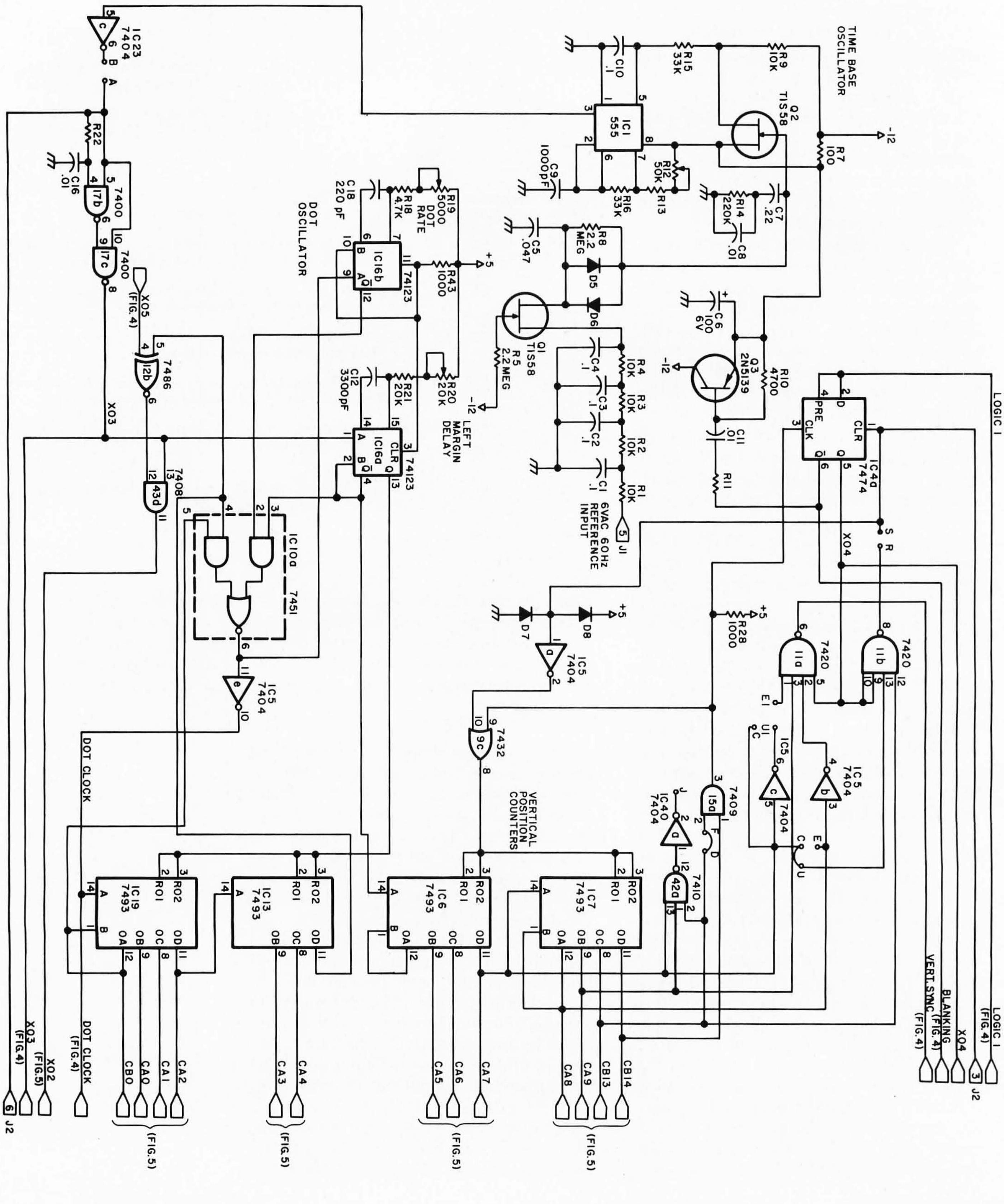


Control Code	Hexadecimal Code	Meaning
0	E0	Inverted Screen. After this command is received, the display will show black cells or characters on a white background.
1	E1	Normal Screen. After this command is received, the display will show white cells or characters on a black background.
2	E2	Disable Characters. Remove information generated by the CT-1024 inputs (if any) from the display.
3	E3	Enable Characters. Allow information generated by the CT-1024 inputs (if any) to be merged in the display output.
4	E4	Enable Graphics. Allow information generated by the GT-6144 display circuitry to be merged into the video display output.
5	E5	Disable Graphics. Remove information generated by the GT-6144 from the video display output.
6	E6	Unused.
7	E7	Unused.

Figure 2: Data Formats for the Control Commands. The unused states of the vertical command format described in figure 1 are used to perform control operations. The control format and operations are shown here. Note that in the program of listing 1, the contents of bits 3 and 4 are assigned zero value arbitrarily since they are "don't care" inputs in control commands.

Now let's get back to the horizontal portion of the circuit again. We left off earlier by saying that one shot oscillator IC16a provided an adjustable delay between the horizontal sync pulse and the generation of data to provide a left margin. We also said that an astable oscillator IC16b which is inhibited during this delay phase via IC10a is the dot oscillator which actually clocks off the dots for each line of video which form the rectangle. To continue the account where we left off, potentiometer R19 sets the cycle time for this oscillator from 0.5 to 1 μs which in turn sets the horizontal width of the rectangles displayed. The "dot clock" output, however, is not the output of IC16b but rather the output of the AND-OR-INVERT gate IC10a. Its output is normally high, but goes low for about 30 ns each time IC16b resets. The 30 ns pulse time is set by the propagation of IC16b and IC10a and is

Figure 3: Timing Logic. This figure shows the design of the video timing chain which governs the operation of the GT-6144. The original diagram of the circuit has been partitioned into three figures for this article, with connections between figures indicated by symbolic names and references to the connecting figure. Power connections for the integrated circuits in figures 3, 4 and 5 are listed in table 1, and the three jacks used for interconnection are summarized in table 2.



very hard if not impossible to see with most oscilloscopes. This dot clock is used to toggle flip flop IC4b which regulates the transmission data from memory to the video output. When IC4b is clocked by the dot clock, the data input on pin 12 from the memory is latched until another clock pulse appears.

The second major portion of the GT-6144 (figures 4 and 5) consists of the memory and its associated components. When no data is being loaded into the terminal by a computer, appropriate outputs from both the horizontal and vertical counters are used as address lines for the memories. IC18 and IC37 along with the necessary inverters form the decoders that select the appropriate memory integrated circuit (ICs 27, 28, 29, 31, 32 or 33) depending on the location being addressed. Note that IC31 to IC33 can be enabled when CB0 = 0 while IC27 to IC29 can be enabled when CB0 = 1, therefore adjacent rectangles alternate back and forth between the two halves of memory. The data outputs of IC27 to IC29 are "wired OR" as are the outputs of IC31 to IC33. AND-OR-INVERT gate IC10b selects which data will be transmitted to the video output depending on the state of CB0.

The desired display for our graphics terminal is 64 cell positions across and 96 cell positions down. This gives a total of 6144 possible character positions which means that at least 13 address lines are needed since $2^{12} = 4096$ and $2^{13} = 8192$. Since some computers are not capable of sending out a 16 bit word, the most logical thing is to use two 8 bit words with some type of temporary storage in the terminal. The input control logic of figures 4 and 5 serves this purpose. The temporary storage in the GT-6144 is done in flip flops IC34b and IC41. Since the GT-6144 can accept data faster than most small computers can send it, only a DATA READY output from the computer is needed. This DATA READY line should be normally high and go low only after the data output from the computer is valid. During the time that the DATA READY line is low one of two things can happen. If the bit pattern from the computer is such that it is a horizontal position (bit 7 is 0) IC30b will trigger, causing the data present to be latched into the flip flops. During this time the memory is unchanged. If the data present is a vertical position (bit 7 is 1) IC30b will not change state but IC39a or IC39c will, depending on the state of the latched low order bit 0 on pin 5 of IC41b. If IC41b pin 5 is 0, IC39c will go low. When the counters come around such that CB0 is 1 (the opposite side of the memory currently accessed by the video timing chain), IC40c will change state causing the data selectors IC24 to IC26 to select the address lines formed by the data from the computer rather than from the counters. IC18 will select and enable the correct memory integrated circuit and after a small time delay caused by C13, R31 and IC15c, a low write pulse will be routed to the enabled memory. When the DATA READY goes back to its high state the write pulse will immediately disappear and after a small delay caused by R36, C17 and IC38b, the memory circuit will be disabled and the data selectors will return to addresses taken from the terminal's video timing chain. Note that the data is written in one half of the memory while the terminal reads and outputs data from the other half. The other half of memory works exactly as above but has B0 equal to 1.

Text continued on page 54

Table 1: Power Connections for Integrated Circuits. This table contains a complete listing of the power connection pins for each integrated circuit in figures 3, 4 and 5.

No.	Device	+5 V	GND	−12 V
IC1	555	8	1	
IC2	74279	16	8	
IC3	7451	14	7	
IC4	7474	14	7	
IC5	7404	14	7	
IC6	7493	5	10	
IC7	7493	5	10	
IC8	7408	14	7	
IC9	7432	14	7	
IC10	7451	14	7	
IC11	7420	14	7	
IC12	7486	14	7	
IC13	7493	5	10	
IC14	7442	16	8	
IC15	7409	14	7	
IC16	74123	16	8	
IC17	7400	14	7	
IC18	7410	14	7	
IC19	7493	5	10	
IC20	74157	16	8	
IC21	74157	16	8	
IC22	74157	16	8	
IC23	7404	14	7	
IC24	74157	16	8	
IC25	74157	16	8	
IC26	74157	16	8	
IC27	2102-1	10	9	
IC28	2102-1	10	9	
IC29	2102-1	10	9	
IC30	7402	14	7	
IC31	2102-1	10	9	
IC32	2102-1	10	9	
IC33	2102-1	10	9	
IC34	7474	14	7	
IC35	7400	14	7	
IC36	7404	14	7	
IC37	7410	14	7	
IC38	7432	14	7	
IC39	7410	14	7	
IC40	7404	14	7	
IC41	74174	16	8	
IC42	7410	14	7	
IC43	7408	14	7	

Table 2: Connector Pin Designations. There are three connectors, labelled J1, J2 and J3 in the figures. The following are pin assignments for these connectors.

J1: Power Jack

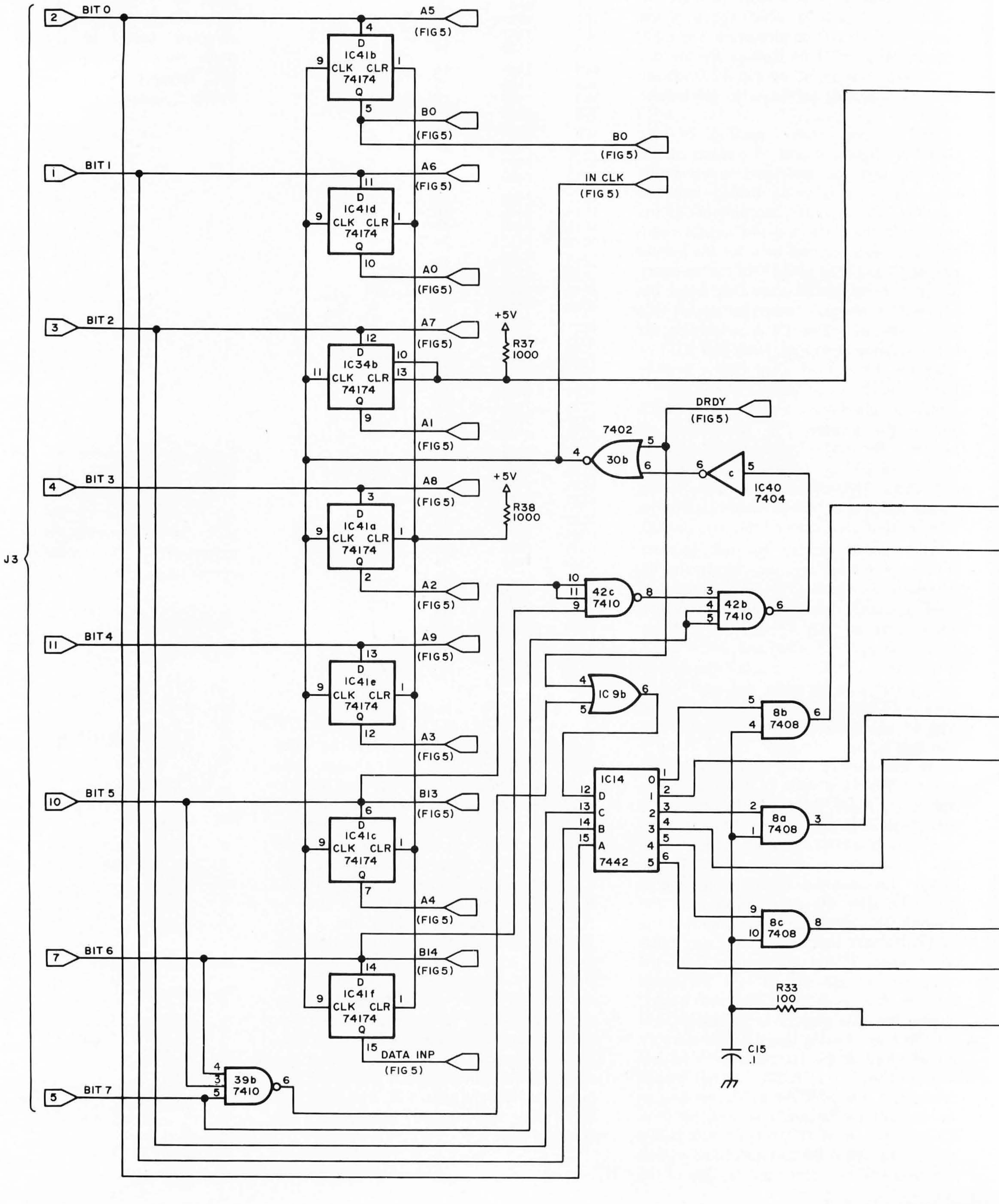
Pin	Assignment
1	−12 V
2	+5 V
3	GND
4	–
5	6 VAC 60 Hz reference
6	GND

J2: CT-1024 Connections

Pin	Assignment
1	GND
2	CT-1024 Data
3	Reset
4	GND
5	GND
6	Sync

J3: Parallel Interface

Pin	Assignment
1	Bit 1
2	Bit 0
3	Bit 2
4	Bit 3
5	Bit 7
6	Data Ready
7	Bit 6
8	–
9	–
10	Bit 5
11	Bit 4
12	GND


J3
BIT 0
BIT 1
BIT 2
BIT 3
BIT 4
BIT 5
BIT 6
BIT 7
IC41b 74174
IC41d 74174
IC34b 74174
IC41a 74174
IC41e 74174
IC41c 74174
IC41f 74174
A5 (FIG 5)
B0 (FIG 5)
A6 (FIG 5)
A0 (FIG 5)
A7 (FIG 5)
A1 (FIG 5)
A8 (FIG 5)
A2 (FIG 5)
A9 (FIG 5)
A3 (FIG 5)
B13 (FIG 5)
A4 (FIG 5)
B14 (FIG 5)
DATA INP (FIG 5)
IN CLK (FIG 5)
DRDY (FIG 5)
+5V
R37 1000
R38 1000
7402 30b
IC40 7404
42c 7410
42b 7410
IC 9b
8b 7408
8a 7408
8c 7408
IC14 7442
39b 7410
R33 100
C15 .1

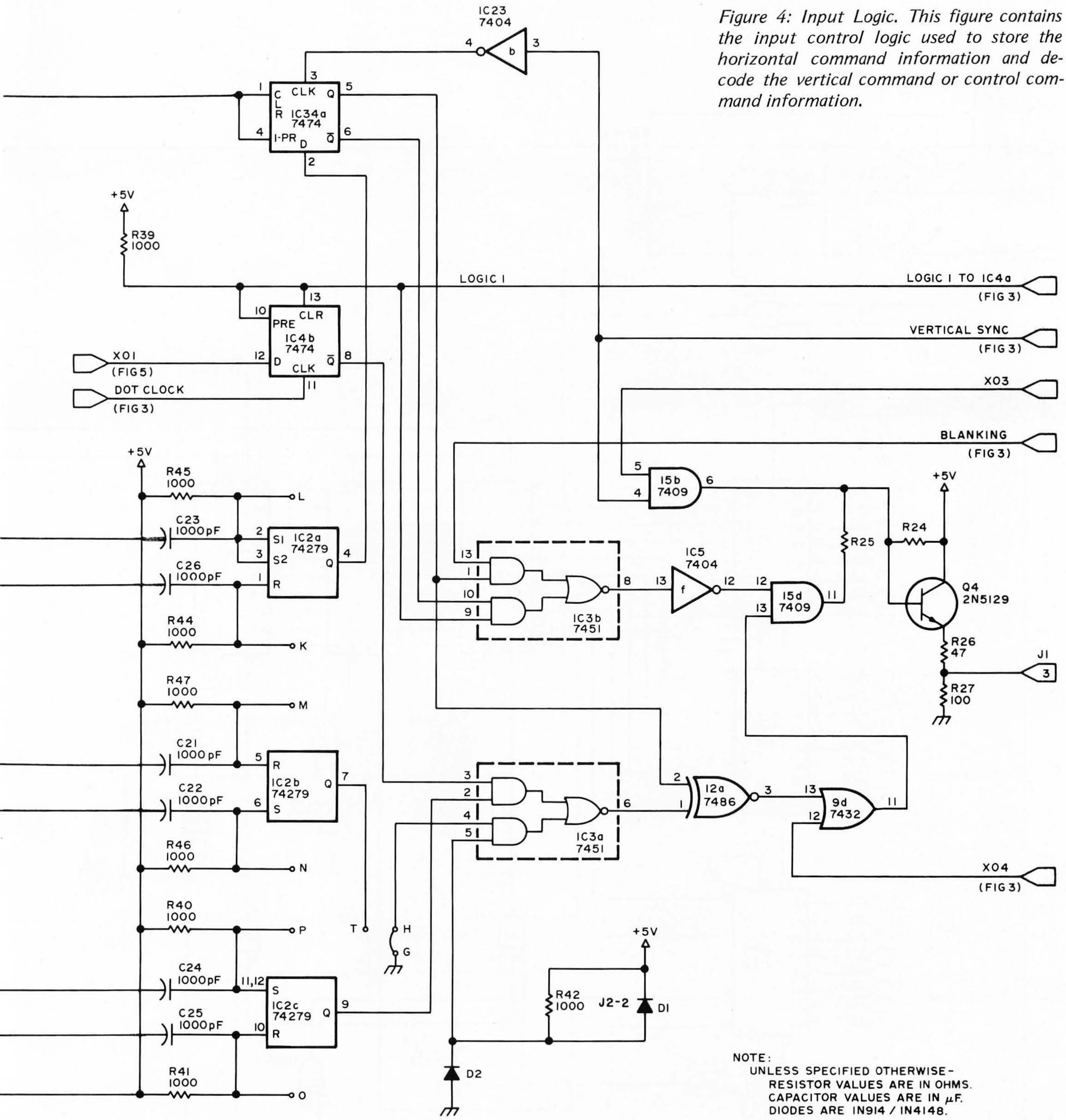
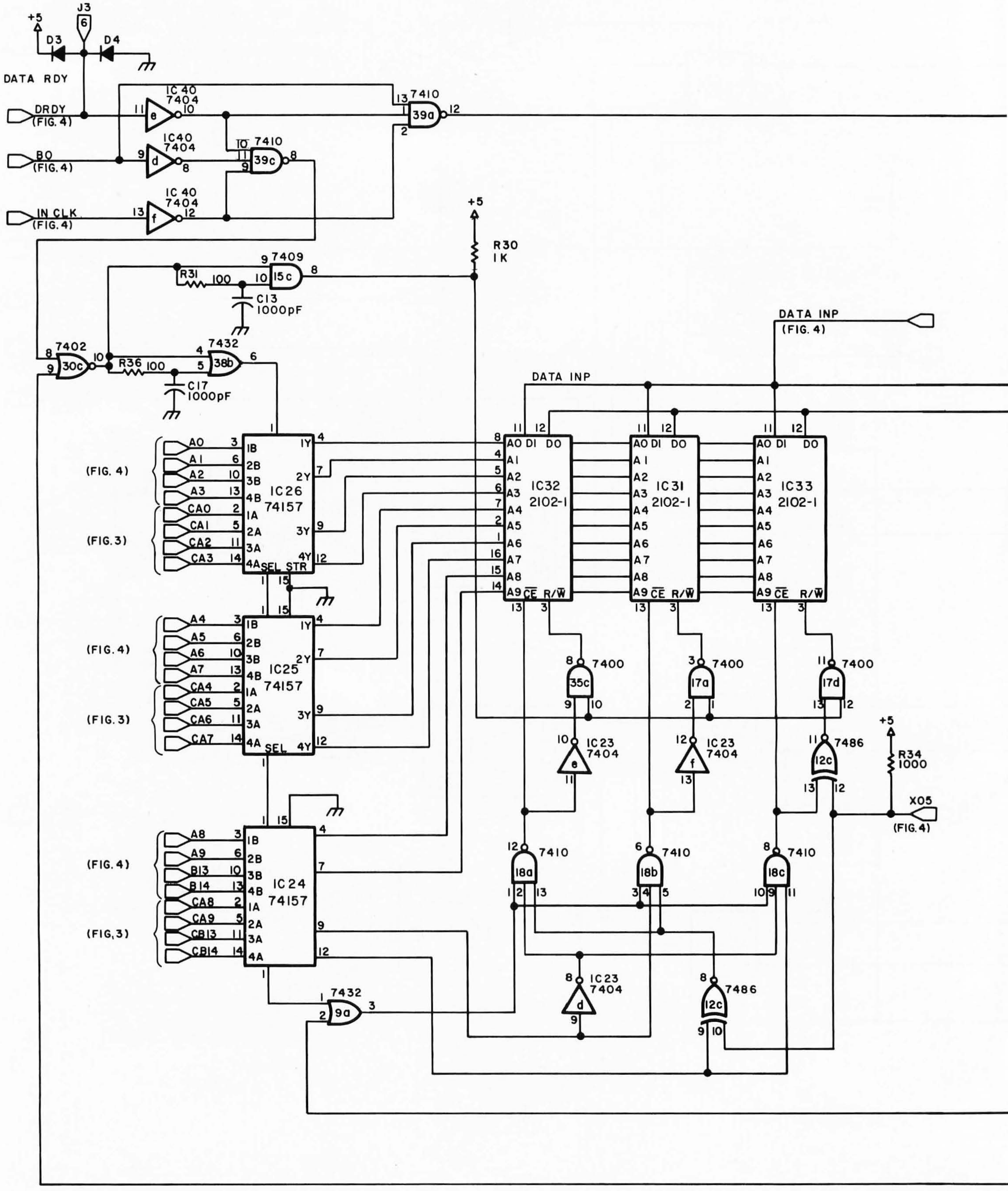
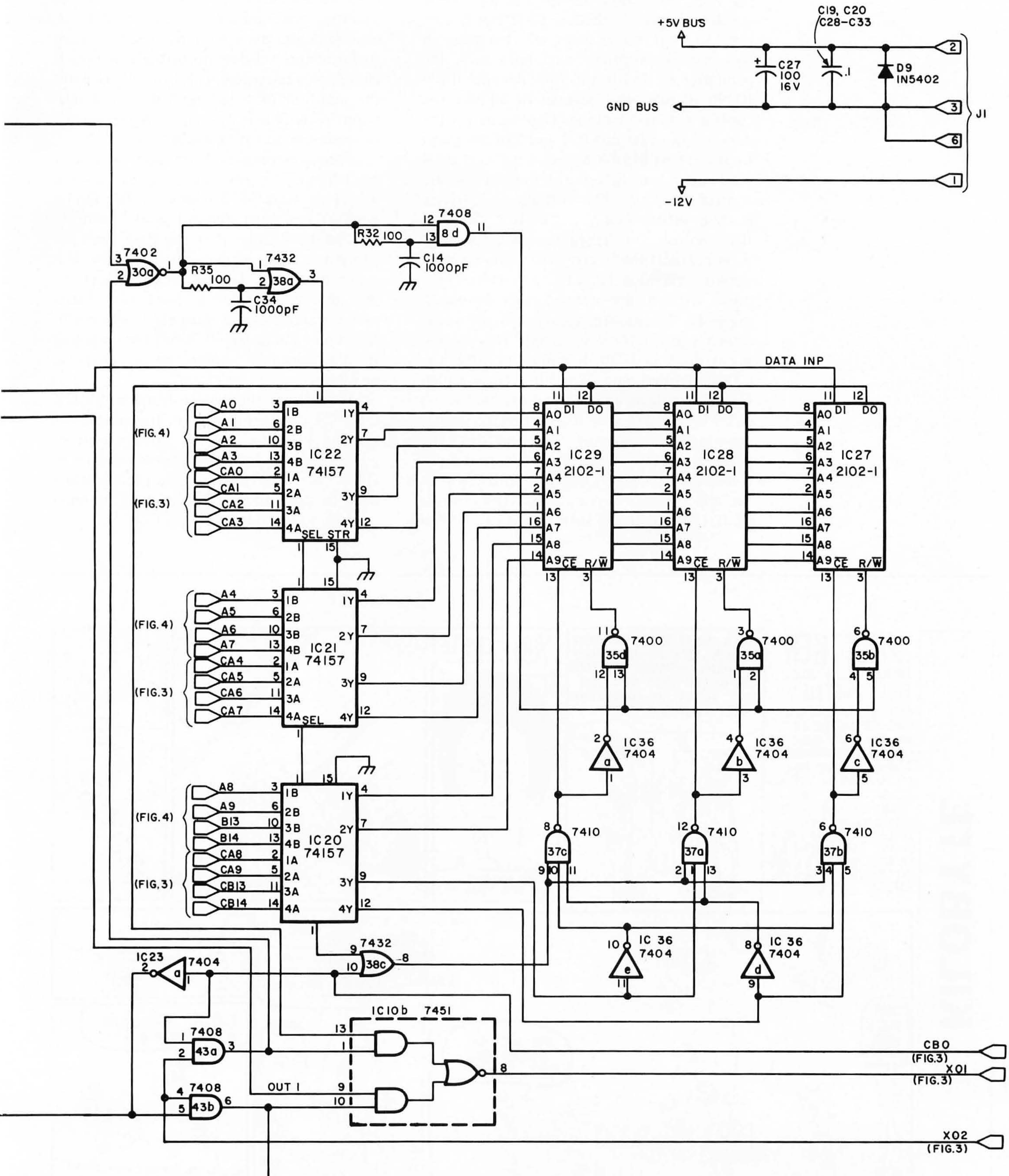


Figure 4: Input Logic. This figure contains the input control logic used to store the horizontal command information and decode the vertical command or control command information.

Figure 5: Memory Logic. This figure contains the logic associated with the memory of the GT-6144. The 6144 bits of information used for the graphic output are contained in six 2102 memory integrated circuits.





+5V BUS
C19, C20
C28–C33
C27
100
16V
.1
D9
IN5402
GND BUS
J1
-12V
7402
30a
R35
100
C34
1000pF
7432
38a
R32 100
C14
1000pF
7408
8d
DATA INP
A0
A1
A2
A3
CA0
CA1
CA2
CA3
(FIG. 4)
(FIG. 3)
IC22
74157
SEL STR
A4
A5
A6
A7
CA4
CA5
CA6
CA7
IC21
74157
SEL
A8
A9
B13
B14
CA8
CA9
CB13
CB14
IC20
74157
IC29
2102-1
IC28
2102-1
IC27
2102-1
DI DO
CE R/W
7400
35d
35a
35b
IC36
7404
7410
37c
37a
37b
IC 36
7404
IC23
7404
7432
38c
7408
43a
43b
IC10b 7451
OUT 1
CB0
(FIG.3)
XO1
(FIG.3)
XO2
(FIG.3)

Text continued from page 49

When the computer sends out data where the high order bits B5, Bc and B7 =1(the control command mode), IC42 pin 8 goes high so that no changes will be made in memory. Appropriate data lines from the computer are fed to the 7442 decoder IC14. IC39b disables the decoder in all but the control command mode. Depending on the data in low order bits 0, 1 and 2 of the input to the GT-6144, the proper output of IC14 is brought low to control the appropriate control function. The outputs of IC14 are latched in the 74279 circuit IC2. Pin 4 of IC2 controls the screen reversing function. IC34 prevents the screen from being reversed except immediately after a vertical sync pulse. While in the reversed mode the blanking pulse is removed, giving a totally white screen (except for characters). This switching is done in IC3b. IC2 pin 7 controls the CT-1024 video data. When IC3 pin 4 is high CT-1024 data is enabled and disabled when pin 4 is low. Pin 9 of IC2 performs a similar enabling or disabling function for the graphics video data. Video data from IC3 pin 6 is routed to IC15d where it is mixed with the appropriate blanking pulse. The outputs of IC15b and d are then mixed in 2N5129 transistor Q4 to give the required composite video output.

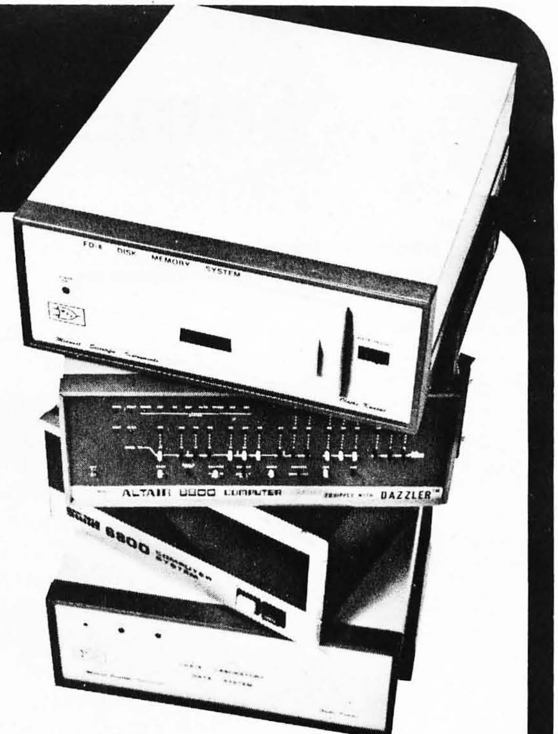
As shown in figures 3 to 5 the GT-6144 unit should work on a video monitor or modified televison set. Because of rigid FCC requirements, the circuit has been designed to be connected directly to the video input circuit of a television. This typically requires the addition of a jack and if normal television reception is desired as well, a switch to select the operating mode.

The data outputs from your computer must be at TTL low state for a logical 0 and TTL high state for a logical 1. The DATA READY line must stay at logical 1 normally and go to 0 only after the data from the computer to be loaded into the terminal is guaranteed valid. The DATA READY strobe should stay low for at least two microseconds and the data must stay valid during this time. Cable lengths from the computer to the terminal should be as short as possible.

If desired the video from a SWTPC CT-1024 can be mixed with the video from the GT-6144 to give an alphanumeric or graphic display (or both) on the same TV or video monitor. No extensive modifications need be made to the CT-1024 but four wires do need to be added through J2.■



No Matter How You Stack Them Ours Comes Out On Top!



OUR FD-8 FLOPPY DISK SYSTEM interfaces to anybody's microcomputer system via a single PIA chip!

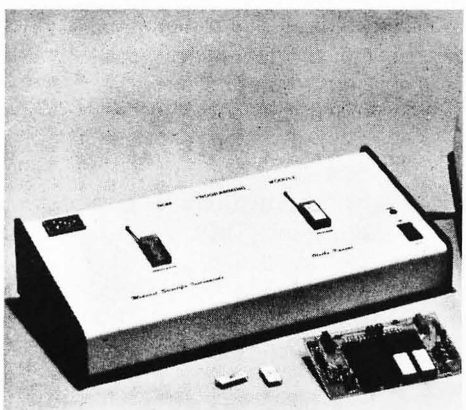
If you don't use PIA's, then one 8 bit bidirectional data port and one output only control port is all that's required — that simple! If you don't have a bidirectional port then separate input and output ports will do.

Full sector buffering in 3K of RAM contained on the controller card itself eliminates any dependence upon processor speed.

Each drive is contained in its own cabinet with power supply. Up to 4 drives may be daisy chained together and selected under software control from a single controller card.

Both single and double density, single or multiple drive units are available.

Complete F DOS software for both 8080 and 6800 systems is provided at no additional charge, including: disk driver subroutines, variable length file management system, disk assembler/editor, and integration with basic.



MSI introduces the PR-1 PROM Programmer and verification module for use with microprocessor systems.

The PR-1 interfaces to any microcomputer system via a single PIA chip. The unit is designed to program 1702A PROMS. Complete software for PROM programming is provided with the system at no additional charge.

MSI software products including our mini assembler, disassembler, and basic are now available on KC standard cassettes. Please specify either paper tape or cassette when ordering.

For the past 6 years MSI has been a leading manufacturer of microcomputer data handling systems for medical laboratories, so we're not new to the business. Four years ago we introduced the first floppy disk system for programmable calculators, which is still in production today. We manufacture CRT terminals, PROM programmers, and a large selection of instrumentation interfaces.

For more comprehensive product information, write MSI at the address below. Incidentally, our products are ready for immediate delivery.

Master Charge & BankAmericard orders welcome.

Here are two MSI Dealers, who can show you our products in action...

Microcomputer Systems, Inc.
144 S. Dale Mabry Ave.
Tampa, Florida 33609
(813) 879-4301

American Microprocessors,
Equipment & Supply Corp.
Chicagoland Airport, P.O. Box 515
Prairie View, Illinois 60069
(312) 634-0076



MSI

Midwest Scientific Instruments

220 WEST CEDAR, OLATHE, KANSAS 66061 • PHONE 913 764-3273 • TWX 910 749 6403 (MSI OLAT)

Some Graphics Background Information

Ira Jay Rampil
University of Wisconsin – Madison
Department of Electrical and Computer Engineering
1425 Johnson Dr
Madison WI 53706

Computer graphics has come a long way since its inception as a laboratory curiosity. It is now a widely used tool for human information digesters. There is the old saying: "One picture is worth one thousand words" and thus information packed into computer generated pictures is now used by such diverse people as engineers, doctors, business people, and choreographers. We are now at the point where TV compatible graphics displays are affordable by hobbyists, experimenters, and other randomly assorted computer freaks.

Graphics hardware has also evolved far from the early dedicated processors and point by point plotting oscilloscopes to today's high resolution plasma displays (the so-called flat TV screens), computer generated holograms, and intelligent multicolor vector displays. The vast majority of commercially available graphics, however, is still based on some kind of single color cathode ray tube. Even among the monochrome displays there are two different families of design, the storage tube design, and the continuously refreshed design. The purpose of this article is to present some background information on the philosophies represented by these two types of CRT systems, since they are important differences and have a large effect on the type of graphics each is capable of producing.

Storage Tube Graphics

Storage tube techniques can well be described as drawing on a paper pad with indelible ink. A line, once drawn, cannot be erased, except by tearing off and throwing away the entire sheet of paper. In other words, once completed, individual sections of a picture (also called subpictures, or glyphs) cannot be electronically deleted or moved without destroying and then redrawing the entire picture. The need to redraw the entire screen might not be so bad, if it were not for the "erase flash," and the usually slow speed with which the graphics terminal communicates with the computer. The erase flash is a bright wink that always occurs when you wipe off the screen of a storage tube. The slow speed is a mixed blessing, actually. These terminals are almost always used remote from the computer, and use simple asynchronous serial data lines, usually around 300 baud, to talk to the computer, just like a raster scan video display terminal. Using an ordinary telephone line for cheap communication, it can take several minutes to draw a reasonably complex image at 300 baud. At the very least, animation graphics, like Space War or LIFE, would prove cumbersome, and disappointing due to communications line limitations. These terminals are best suited for high resolution still lifes such as charts and schematics. Accessories to produce hard copy from the screen are usually available. Storage tube terminals have the advantage of being quite useful at great distance from the processor. They are also comparatively simple in design and cheap to build. The exception is the price of the storage CRT, which can run over $1000 in "onesies" for a large high resolution tube, probably explaining why they are not often seen in amateur computing circles.

Continuous Refresh Graphics

The continuously refreshed or graphics processor type of display is like the storage tube display only in the way in which it draws a line or vector. There the similarity ends. In this system, vectors disappear almost as fast as they are created. An image appears of constant intensity because it is redrawn or refreshed many times a second. This illusion, called flicker fusion, is the same principle by which the TV and movie industry simulate continuous motion. If an image flickers fast enough, the brain perceives it as continuous. Obviously, if there are many lines in the image, there has to be a very fast link between the display and the memory where the image data resides, in order to refresh the screen in the usual 1/30 second (or 1/60 second for TV without interlace). Since drawing a previously created image is just a matter of shipping the picture information off to the display, and therefore requires no great amount of com-

Perhaps the most dramatic example of the need for high resolution graphics is demonstrated by these pictures of the famous "original" lunar lander program written for the Digital Equipment Corporation GT-40 and supplied with that unit as a demonstrator. These pictures were made by the author using University of Wisconsin facilities, and illustrate a couple of typical frames from the interactive graphics version of lunar lander.

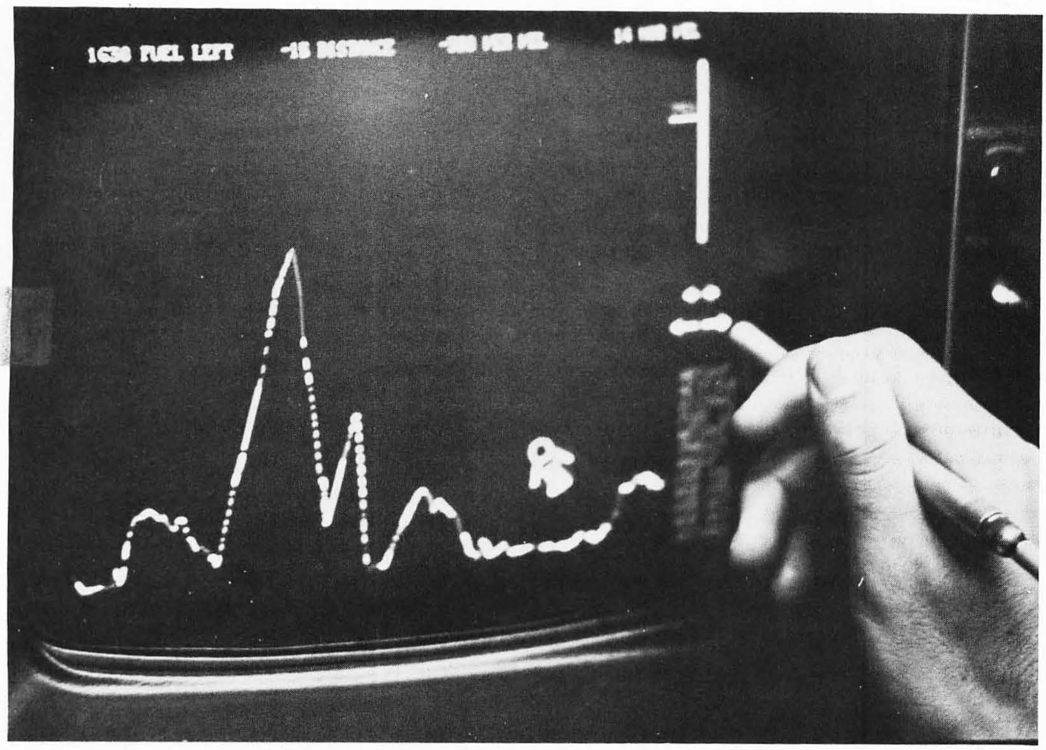


Photo 1: The lunar module is controlled by a light pen, and is seen initially high above a lunar plane.



Photo 2: Eventually, if all goes well, the pilot (with his or her magic wand control stick) lands the lunar module near the golden arches of a famous fast food chain, orders some food, takes off and leaves.

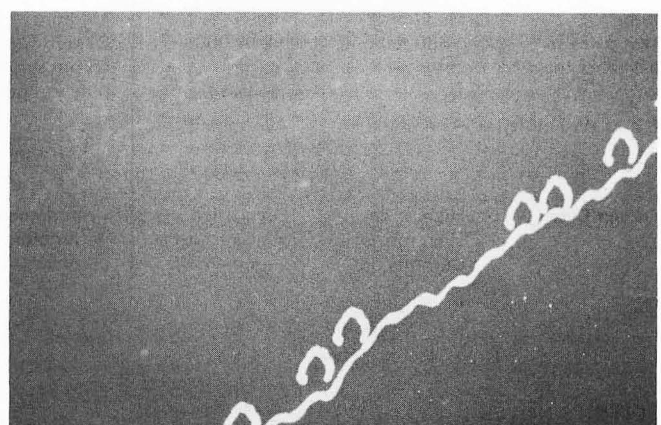


Photo 3: Of course, if mistakes occur, there is a crash. This picture shows the flying lunar lander above a lunar mountainside with "rocks" and the remains of previous crashes of the module by inept pilots.

With the 256 by 256 graphics capabilities of the newer display output devices for televisions, the personal computer is getting close to the resolution needed to do the lunar lander simulations shown here, at a far lower price than was previously possible.

Table 1: Some graphics display devices listed for comparison.

Feature	Professional: Tektronix 4006-1	Professional: DEC GT-40	Systems of interest to individuals: ISC Intecolor	SWTPC GT-6144	Cromemco TV Dazzler	MiniTerm Assoc MERLIN	Matrox
Design Type	Storage tube	Refresh vector	Raster scan	Raster scan	Raster scan	Raster scan	Raster scan
Interface	Bit serial	DMA	DMA	Byte parallel	DMA	DMA	DMA
Point Plotting	Hardware	Hardware	Hardware	Hardware	Hardware	Hardware	Hardware
Vector Graphics	Hardware	Hardware	Software simulation	Software simulation	Software simulation	Software simulation	Software simulation
Resolution	High 1024 x 780	High 1024 x 768	Low 160 x 100	Low 96 x 64	Low 128 x 128 (BW) 64 x 64 (color)	Low 80 x 100 160 x 100	Medium 256 x 256
Drawing Speed Limitations	Communications interface limited	200 μs per vector (fixed)	Byte transfer speed	Programmed IO for each point	Byte transfer speed	Byte transfer speed	Byte transfer speed
Character Generation	Hardware	Hardware	Hardware	Software	Software	Hardware	Software
Blinking	Not possible	Hardware	Hardware	Software	Software	Software	Software
Variable Brightness	No	8 levels	No	No	No	No	No
Remote Operation	Yes	Yes—contains its own PDP-11/05	Intelligent terminal use	Yes	No	No	No
Altair Bus Plug in?	No	No	No	No	Yes	Yes	No
Multicolor?	No	No	Yes	No	Yes	No	No
Is Animation Possible?	No	Yes	Yes	Yes	Yes	Yes	Yes
Hard Copy (Not Photos)	Yes (extra cost)	Yes	No	No	No	No	No
Built-in Display?	Yes	Yes	Yes	No	No	No	No
ROM Software in System	Not applicable (used with large systems)	Only a bootstrap	Yes System monitor	No	No	Yes MERLIN monitor	No
Price (Kit price if available that way)	Medium ~$3000	High ~$14,000	Medium $1,395 (includes computer, display keyboard)	Low $98.50	Low $215	Low $249 + $34.95/ROM	Low-Medium $630
Kit or Assembled	Assembled	Assembled	Kit	Kit	Kit	Kit	Assembled (requires bus interface)
Design Class	High price High resolution Commercial	High price High resolution Commercial	Complete system with color graphics	Low price Low resolution Stand alone display generator	Integral plug in for color graphics	Integral plug in with systems software	Medium resolution black box

EDITOR'S NOTES

This table is by no means complete. It is intended to cover a spectrum of different systems with different prices and different user orientations. If each column is considered to be a class of product, then the device shown is a typical member of that class. The rationalizations for selection are roughly as follows:

- The DEC GT-40 and the Tektronix 4006-1 are included for comparison purposes only to contrast prices of commercial equipment intended for use with large scale installations. The GT-40 virtually requires a minicomputer, and is a very flexible (if now slightly outdated) system widely used by graphics hackers at university and research institutions. The Tektronix product and related products are within the range of a moderately wealthy private computer hacker, but are hardly products for mass consumption. The 4006-1 is also widely used in the university and research communities.
- The ISC system is an example of a complete system — computer, video display (color) keyboard, and systems software in ROM — which is available in kit form at a price within the range of an individual.
- The Southwest Technical Products GT-6144 is a low resolution, low price stand alone product, intended to be absolutely the least expensive approach to the graphics output problem. The intention is quite well fulfilled. It does require a television or monitor, however, as do all the amateurs' products except the ISC system.
- The Cromemco TV Dazzler is the first Altair bus compatible product available to amateurs at a reasonable

puting, we needn't bother the main processor. Instead, we offer the display direct access to the memory where the image is kept. There are two ways to do this; the first is to allow the display to cycle steal, or have "direct memory access." This is the way many high performance commercial systems, like the DEC GT-40, work. This method does not allow the display to be far removed from the computer and its memory. In the GT-40, for example, the input to the display is about six inches from the memory board.

The other approach is to give the display its own memory for the image, and its very own simple processor to keep track of what is going on with the display. The central processor then is allowed access to the display memory, and need only concern itself with the display when it wants to change the image. Thus the display can cycle through its own memory as it pleases, and the CPU can modify the image at a rate determined by economy and necessity. This is the method used by the typical homebrew or commercial video display used by BYTE readers. The local memories which have been used in these devices range from delay lines, shift registers, and programmable random access core or semiconductor memories for alphanumerics, up to dedicated disks for multicolored high resolution applications which require vast amounts of storage.

The primary advantage of continuously refreshed graphics is high speed. This is the mode in which animation work is most often done. Since the images are stored in memory they are very easily manipulated at high speed. The resolution can be made equal to or better than storage displays.

The currently popular personal systems displays use the refresh graphics method in combination with a video raster scheme. Instead of directly drawing vectors on a screen, they can only mark points that lie along the closely spaced parallel horizontal lines that make up a normal TV raster. The effect is to limit the maximum resolution to about one third that of commercial vector displays. There is also an additional software burden on the processor whenever it is desired to simulate vector and character hardware.

The chart accompanying this article as table 1 summarizes the characteristics of several representative graphics display devices. Take this table with a grain of salt, though, because today's specs are tomorrow's history. I expect that the features of amateur raster scan devices will eventually come close to the commercial continuous-refresh display, except, of course, for price.■

price which could produce color graphics in limited 64 by 64 resolution. Installation is a simple matter of plugging it into an unused Altair (or equivalent) computer slot.

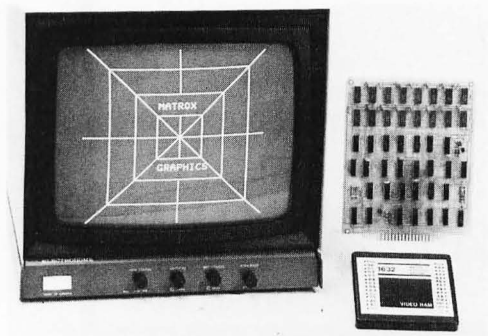
- The MiniTerm MERLIN system is an example of a unique combination of character generator hardware, dual resolution black and white graphics, direct memory access to main memory, and systems software in ROM form. It uses existing memory of an Altair compatible computer as its graphics refresh source.
- The Matrox display was not originally intended for the personal systems markets. It is not directly plug compatible with any particular computer, although interfacing to a bidirectional microcomputer bus is a straightforward matter. It has been promoted in engineering trade magazines as an "OEM" product. Circa August 1 1976 it is the only commercially available option for such high resolution at such a low price (Space War hackers take note). For further information, see "What's New?" in this issue.

What's New?

Graphics Display Memory

Matrox Electronic Systems, POB 56, Ahuntsic Stn, Montreal, Quebec H3L 3N5 CANADA, (514) 481-6838, has introduced this exciting new peripheral module for use in generation of graphics outputs. The module is the MTX-256**2 graphics display memory, shown here along with the model 1632 video display memory.

The MTX-256**2 graphic display is a modular low cost graphic system designed for microcomputer applications. It displays a total of 65,536 dots arranged in a 256 by 256 dot raster with each dot individually addressed under program control in less than 3.4 us.



The system consists of two modules. A central timing unit module (CTU) provides all television synchronization signals and interface timing signals for any microprocessor and for image memory (IM) module. The image memory is a 65,536 bit programmable memory, and it stores one television frame (256 by 256) with one level of intensity. No external refresh is required. The image memory is continuously scanned to produce a video signal that drives any standard monochrome or color television monitor directly.

Expandability

Using one CTU and one or more IM modules, various graphic systems can be designed by simple interconnections. Gray scale, color, image motion, light pen, cursor plot, vector plot, point plot, bar graph plot, alphanumerics and ROM screen patterns are easily implemented.

The MTX-256**2 provides separate horizontal and vertical sync outputs which can be used to drive a slave MTX-1632 SL video programmable memory to obtain fully alphanumeric graphic capability, as shown in the photo.

The simplest graphic system (MTX-256**2-1) consists of one CTU module and one IM module. The system is also available as a single 6.5 by 6 inch (16.6 by 15.2 cm) printed circuit board, +5 V (400 mA); +12 V (120 mA). This unit will provide a 256 by 256 dot raster with one level of intensity (0 . . black, 1 . . white). It directly connects to a microprocessor data and address bus and no additional hardware is necessary.

To a microprocessor an MTX-256**2 looks like a write only memory with four 8 bit locations. The locations with addresses 1 and 2 are X and Y coordinates of a given dot. Cursor (dot address) is loaded by writing into X and Y address (8 bit each).

The intensity (color) is loaded by writing into location 0. Existing cursor address is first advanced (moved to the next of 8 possible directions, specified by 4 MSB bits). The remaining bits specify the intensity of the addressed dot. Location 4 is used for added intensity or color resolution (up to 24 bits for each dot). Clear and present screen inputs are also available (minimum screen pulse width is 12 ms).

Output is a standard composite video signal (75 ohm, crystal controlled). Both American (60 Hz rate) and European (50 Hz) versions are available. Price is $598 in unit quantities for the basic MTX-256**2-1. Delivery is 2-8 weeks.

Editor's Note:

In conversation with Lorne Trottier of Matrox, at the Personal Computing 76 show in Atlantic City, I learned of Matfox's intention to create a variation of the display described here which will plug directly into the Altair/IMSAI/etc bus [nested note: unofficially grapevined as the "S100" bus to eliminate the kluge of listing all the names of all the processors using it . . . CH] . . . CH]

Product Description:

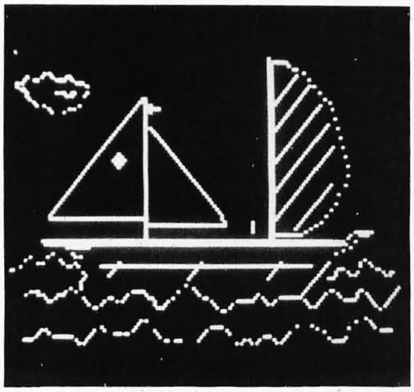
The MERLIN Video Interface Adds a Visual Dimension to Your Altair or IMSAI

What is missing when you purchase a main frame computer from the catalogues of MITS Inc or IMS Associates? Asking that question has recently started several companies purveying products to the small systems market. The identification of gaps in a product line, then independently filling these gaps is a highly respected tradition in the computer field, a tradition which has hardly changed as the market widens through microcomputer technology. An example of a creative and innovative product idea which augments and enhances an existing computer product line is provided by the new MERLIN board manufactured by MiniTerm Associates, Bedford MA. The following article is edited from materials supplied by MiniTerm.

MERLIN (trademark of MiniTerm Associates) is a new concept in peripherals modules for mainframe microcomputer systems. It is a combined hardware and software package which provides a keyboard video interface with 20 lines of 40 ASCII encoded characters, graphics with a resolution of 100 by 160 points, and read only memory software for Monitor and Editor programs. The entire price of this unit, $249 plus $34.95 for the ROM software, is well within the budget of many small systems users. In fact it is hard to find a combination of the features MERLIN has which even approaches the cost effectiveness of this two card Altair and IMSAI compatible plug-in system. Output is an EIA standard video signal to drive a TV monitor or modified television set.

MERLIN'S two boards provide far more than just an ASCII and graphics interface. A sophisticated DMA controller allows MERLIN to display information from any part of the computer memory. Also, since the display starting address is software controlled, the memory segment displayed can be changed with every screen refresh.

The number of ways in which the memory selected can be displayed boggles the mind. And since the display mode is software controlled, it too can be changed with every refresh. One useful mode option saves memory by beginning a new line whenever a carriage return is detected – eliminating the need for spaces to the end of a line. This enables typical text or program source code listings to be stored in 30 to 50% savings of memory space depending on the actual text data involved. Another mode selects white or black characters. Control characters can be inverted or blanked – great for storing "invisible" information on screen for some games. Carriage returns are displayed as boxes or blanks and the cursor winks or remains steady. The dense graphics mode enables 2 K words of memory to be displayed as 160 dots horizontally by 100 dots vertically, or if memory is in short supply, 1 K bytes can be used to display as 80 dots horizontally by 100 dots vertically until your next birthday.





If you are worried about loading all those great bytes into your machine with toggle switches, don't. MERLIN provides a parallel input port to which most keyboards may be directly tied and sufficient regulated power to run them. MERLIN also has a serial input port which may be used for binary data from tape signal conditioners, switches, thermocouples or other arcane hobbit paraphernalia. The serial output port is also uncommitted, and the pair can be used to implement a cassette IO interface.

Best of all, you need not be a software genius to use MERLIN. Monitor and Editor functions as well as general purpose subroutines, selectable IO drivers and keyboard and display drivers are also stored in MERLIN's optional on board MBI ROM. (If you are a software genius and have a programmer you can substitute up to 2 K of your own 2708 EROM or 1 K of 2704 EROM.) Provided with the MBI ROM are also 128 words of scratchpad programmable memory to hold MERLIN's vital statistics and Monitor stack. Linkage through this scratchpad to user defined routines creates a system which can be expanded virtually without bound. (As though the ten user defined keystroke executed functions built into the monitor weren't enough!)

Monitor functions provided by the MBI ROM include the ability to fill any portion of memory with a given hexadecimal value, the ability to sequentially modify memory contents, and hexadecimal dumps to the display. Processor registers may also be examined or modified. ASCII text may be input to memory, blocks of memory may be copied, the beginning and end of the display area and display format may be set, and user programs may be executed with up to three breakpoints. The breakpoints enable the user to return control to the Monitor after executing only part of his code. Once in the Monitor the user may examine or modify memory or registers before executing more of his program.

The Editor functions can be invoked by keystrokes or called from user programs. The functions include:

Control of cursor movement (up, down, right, left, home, two tab sets, and tab).

Either characters or lines may be inserted, deleted, or replaced.

Scrolling can be automatic at any line number or under manual or program control. Scrolling may be page up or page down or absolute.

Automatic scrolling can provide wraparound at the end of the defined display area (up to 64 K), so that using the "clear display after cursor position" and "copy block" functions provides an easily implemented "infinite" display area.

Finally, this ROM contains direct linkage to a second optional 2 K ROM and forthcoming cassette-modem interface ROM. MERLIN's second ROM contains extended Monitor/Editor functions plus general purpose graphics control subroutines.

A soon to be announced optional board for MERLIN will increase capabilities to include color graphics, and with 320 horizontal by 200 vertical resolution for black and white graphics. MiniTerm Associates is located at Box 268, Bedford MA 01730. ■

Continued from page 16

run APL, SNOBOL, LISP, PASCAL, ALGOL, TRAC™ etc), and with the falling prices of memory, will do exactly that. I know several personal computer hackers who have pretty nearly filled up a 64 K memory address space. If a person has a choice between a system that costs $1000 with 16 K of memory and the same system with 32 K at an incremental cost consistent with the new 16 K chips' estimated costs, the extra $300 to $500 at the store may be well worth it; and as the technologies improve, more and more memory will be potentially possible for the same dollars.

And there lies the problem. With memory getting cheaper and cheaper, we are running out of address space in the 8 bit LSI processor! The typical 8 bit or 16 bit LSI processor addresses 65,536 bytes and that's all. For many purposes, such as creating data base structures with large buffered data areas internally, it would be extremely convenient to have much larger main memory areas. For instance, in a super text editing system allowing 50 K bytes for buffer areas limits one to about 18 standard manuscript pages in memory at any given time (assuming no paging to expensive electromechanical magnetic media). With memory becoming a very cheap commodity, one would like to have as much of it as possible on line for such applications. Similarly, in setting up information management systems for the home user, the more memory the better, since the resource can be used both to speed up computation by using table concepts, and to minimize IO.

Given this desire for lots of memory, and the limitations of 16 bits, the search for alternatives is bound to occur. One place that this can come out is in newer LSI processor designs. For example, consider the Motorola 6800 as it is presently implemented. It resides in a 40 pin package, but the simplicity of its interfacing requirements leaves two pins unused at present. In a hypothetical "improved" version, the design could be modified to support 18 bits of addressing, or a maximum of 256 K bytes (262,144 bytes) with some changes in the internal architecture. The simplest internal change would be to use the uncommitted bits of the status register (bits 7 and 6) as "bank selection" bits to implement extended external memory by the traditional "bank switching" trick. Of course, it's hardly likely that this design change will happen, so the extension to memory sizes above the address range of the machines will have to be accomplished by the individuals designing the computer products which employ LSI central processors.

Bank switching kluges can be done externally to a processor as well. In a personal computing system with ROM software of compilers or interpreters, one way would be to partition a 32 K segment of address space into two 16 K regions along the boundaries of the 16 K chips, with bank switching used to alternate such extended regions of memory. Again, in a hypothetical case, consider a 4 bit mapping vector for each 16 K half of this region of addressing. The 4 bit bank selection could point to 16 banks of 16 K bytes or 256 K bytes, with the core of 32 K bytes retained for system monitor and IO functions. But these are just conjectures of ways to accomplish the extension of mem-

Articles Policy

BYTE is continually seeking quality manuscripts written by individuals who are applying personal systems, or who have knowledge which will prove useful to our readers. Manuscripts should have double spaced typewritten texts with wide margins. Numbering sequences should be maintained separately for figures, tables, photos and listings. Figures and tables should be provided on separate sheets of paper. Photos of technical subjects should be taken with uniform lighting, sharp focus and should be supplied in the form of clear glossy black and white or color prints (if you do not have access to quality photography, items to be photographed can be shipped to us in many cases). Computer listings should be supplied using the darkest ribbons possible on new (not recycled) blank white computer forms or bond paper. Where possible, we would like authors to supply a short statement about their background and experience.

Articles which are accepted are typically acknowledged with a binder check 4 to 8 weeks after receipt. Honorariums for articles are based upon the technical quality and suitability for BYTE's readership and are typically $25 to $50 per typeset magazine page. We recommend that authors record their name and address information redundantly on materials submitted, and that a return envelope with postage be supplied in the event the article is not accepted. ■

ory, given the need for utilizing existing machine architecture.

Enter the Stack Machine

The real reason we need to use more memory than 64 K is our need to use high level language to program sophisticated information constructs. Traditional LSI processor designs to date have merely been adapted versions of widely used designs to the constraints of LSI technologies. A 6800 is simply an 8 bit approximation of what minicomputers have always been, and minicomputers are just miniature versions of the earlier, larger machines. Sure, improvements in architecture have been made, such as use of subroutine linkage stacks with built-in pointers, but all the present LSI processors are cut of the same mold. If we are running out of room in address space with the present designs because of a need to run high level programming languages and intricate information systems written in these languages, why not use the need for larger address space as a superficial spur for a high level language oriented general purpose processor itself?

There is a whole technology of machines oriented towards high level languages, a technology principally seen in the commercial products of the Burroughs Corporation and experimentally in computer science and software engineering contexts. This is the technology of the high level language machine, or stack machine. This is a processor designed to have a machine language which is very close to a high level language consistent with structured programming concepts. This is the world of the computer whose instruction set includes control structures like the IF instruction, the THENDO instruction, the ENDELSEDO instruction, the ENDIF instruction, and data constructs appropriate for a high level language. The grapevine currently jingles with rumors that such a machine will soon be on the market, possibly labelled "Z90", and I know of at least one university laboratory version intended for reduction into LSI masks. High level languages are what make general purpose computers useful to the largest number of people, so eventually I expect to see stack machines in LSI form, intended for general purpose computers.

Use of stack machines sure won't cure the address space saturation problem, but if one has to add some extra bits to the memory address bus, why not include a real improvement in the processor architecture at the same time as the new design is created?■

Authors take note: *A tutorial on the architecture of stack machines, the reasons for using such machines, and what it is like to program such a machine would be excellent article material for BYTE magazine.*■

Publisher's Notes

Personal Computing 76 in Atlantic City:

Something for Everyone

by Virginia Peschke

Personal computing enthusiasts were treated to a huge, successful convention in Atlantic City on August 28 and 29. BYTE will have a complete pictorial report next month, but briefly: More than 80 manufacturers and suppliers had booths to display their latest products and exchange ideas with users; dozens of technical talks and several seminars provided information on every phase of personal computing; and a huge supply of valuable door prizes was distributed to many lucky attendees.

The speakers at the Saturday night banquet were all greatly entertaining. As a grand finale, John Whitney showed his technically amazing and esthetically beautiful movies of animated computer graphics with musical accompaniment. Though everyone had had a full and exhausting day, John's movies held us all spellbound until almost one in the morning, and various persons inspired by the show were engaged in conversation until much later.

For those of us who aren't true computer hackers (I'm a jade carving nut myself) the choice of Atlantic City as a site was indeed fortuitous. Though it's somewhat difficult to get to from some parts of the country, once there the computer hacker is relieved of the usual domestic problem; that is, how to keep the family entertained. Atlantic City offered so much for the people who didn't wish to spend much time at the show that the computer buff was unlikely to hear "We've been here an hour, when can we go?" First, there was the superb sand beach with a warm, gently rolling ocean, the amusement park, the boardwalk train, and the great selection of restaurants and shops. Where else could one find, within several blocks of each other, dealers in crystal paperweights, archaic Chinese jade, oriental rugs, British tweeds, Spanish lace, magic tricks, health foods, and many other things too numerous to list? No, families were not bored and hackers were free to spend all the time they wanted exchanging information.■

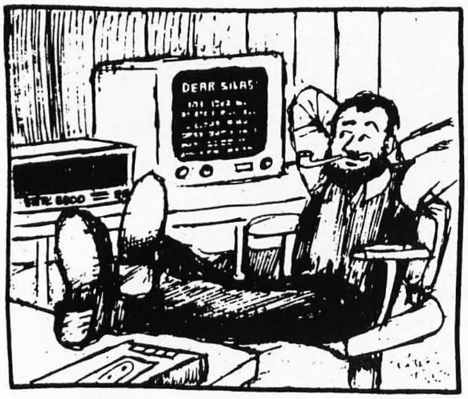
What's New?

Miscellaneous

Based on fliers picked up at the Personal Computing 76 show in Atlantic City, August 28 and 29:

Edityper Systems Corp, subsidiary of Tycom Inc, 26 Just Rd, Fairfield NJ 07006, is manufacturing a do it yourself Selectric typewriter conversion kit. This "easy to install" kit for any IBM Selectric I or Selectric II typewriter "provides quality hard copy output for all microprocessor devices." The kit contains four modules: a selection assembly, function group assembly, shift assembly and tab assembly. It is driven by a 50 pin connector and requires user supplied power of 27 V at 1 A peak, and 5 V at 0.2 A. The price is $395 plus shipping. We'd be most interested in publishing a photo article by a reader who installs this item in a typewriter and integrates it as hard copy output for his or her system.

STM Systems, POB 248, Mont Vernon NH 03057, showed off the new Baby! Floppy Diskette Storage System. This new "mini floppy" drive with power supply and controller will be available in the near future for $750. It uses a miniature floppy disk of about 5 inches (13 cm) diameter and stores 90 K bytes (formatted) on 35 tracks with average latency of 100 ms, 350 ms average access time. The drive was shown in prototype form at the convention.■



Letters

ZEROING IN

I think the point has again been missed in the great 8080/6800 controversy.

Reader Cochran (August 1976) compares the IO structure for what seems to me the millionth time. Really now, what would it take to convert one bus to the other? A couple of AND gates maybe. And who cares if the 6800 has a few less memory locations out of 65536?

The major difference is the purpose behind the two instruction sets. While the 8080 is superior at handling structured data, the 6800 is better at arithmetic. The former has an overloaded accumulator while the latter has an overloaded index register.

The single accumulator of the 8080 requires many transfers to temporary storage. This is most frustrating when doing calculations that involve intermediate results. Unfortunately, the arbitrary nature of the instruction set makes improvement difficult. Zilog's Z-80 is a slight improvement, but only compounds the inconsistencies already present.

The 6800's major fault is its lack of index register capability. However, improvement by adding instructions is easy. There should be an instruction to push the index register onto the stack and one to pop it off. An instruction to compute a 16 bit address by adding the accumulators to the index register is essential for reentrant programming.

Both the 6800 and the 8080 need improvement. Because the 6800 has a cleaner architecture, it has greater potential. So when Motorola comes out with a replacement chip, I'll be first in line. [*Not if I get there first. . .*CH]

Stephen Ma
Glenn Fawcett
5170 James Walk
Vancouver BC CANADA V5W 2K4

Grapevine, Austin TX, has it that the new MC6809 product (the Motorola answer to Z-80) will have: Two 16 bit true index registers, block move, block search, and perhaps even some 16 bit arithmetic to enhance pointer manipulations and address calculations at run time.

A CASE OF DUPLICITY?

Finally ariving home from work after another hard day, seeing my August issue of BYTE in the mailbox immediately calmed my nerves. I settled into my semi-automatic reading chair, placed my copy in the scanner, punched up a beer and sausage combo on the console and prepared to better myself. Greedily I began. Not wanting to flounder about, I immediately flipped to "In This BYTE." The highlights leaped at me. Another fantastic array of information! (Up to this point, I had read every issue of BYTE cover to cover and this one promised to maintain the trend.)

I read to the blurb about the Zilog Z80 and no farther. Expecting to find information to delight and astound me, I started flipping toward the article. One page farther and the ad for the "super chip" whet my appetite and set me flipping furiously. My expectations only rose as I skimmed for page 34 and the "Microprocessor Update: Zilog Z80". Page 25. (Just nine more.) Flip. Flip. Page 32. Flip. Page 50!!! Flip back. Page 32. Flip. Page 50???

I tore my BYTE from the scanner, and manually searched for page 34, but nowhere could I find it. It seems my copy jumps from page 32 to 49, and again from page 80 to 97. The fact that there were doubles of pages 17-32 and 97-112 didn't seem to make up for my loss.

I fell back, dejected, just as my chair served me my beer. But the force of me collapsing backward thrust my beer into the vibrating unit which promptly shorted out, flinging me headlong across the room.

I am now in traction in the county hospital. I am also very dejected because now I have all this time and still can't read about the Z80. Could you find it somewhere in your heart to fill this emptiness.

Deplorably yours

W R MacLeod
GTE Automatic Electric Labs
4501 W North Av
Melrose Park IL 60160

Mr MacLeod received the following reply:

Dear Mr MacLeod

Enclosed you will find a complete August issue of BYTE. We hope you enjoy reading the article "Microprocessor Update: Zilog Z80," during your stay in the county hospital. If it appears to be an extended stay, please let us know if we may forward the September and October issues to you.

The staff at BYTE wishes you a quick recovery.

[Any reader who received a similar glitch should send his/her BYTE back for replacement.]

COMPUTER PHREAQUENCY

BYTE No. 13 (September 76), page 12, letter from Melvon G Hart, W0IBZ, "Attention Hams! . . ." Try 3.865 MHz (LSB) on Thursdays at 2300 GMT. You will find several computer-phreaque/amateurs there.

Carl K Zettner, W5HFG
108 Moss Dr
San Antonio TX 78213

HUMANE COMPUTING?

I am working on a survey of the humanistic use of computers and would like to get in touch with other people who are interested in this field. I want to learn about what can, and is, being done to make the use of computers as humanized as possible. More specifically, I want to know:

- What are the potentially humane applications of computers?
- What are the important ingredients in humanized computer systems?
- What is currently being done in the field of humane computing, both in terms of theory and practice?
- Which individuals and groups are actively doing this work?
- What are their particular goals and objectives?
- What are the results so far?
- What are the major factors determining their successes and failures?

I would like to hear from anyone who can contribute to answering such questions. They should write to me describing their thoughts, fantasies, plans and experiences related to the humane use of computers. Besides corresponding with people, I hope to actually visit and talk to as many contacts as possible. All the material I collect in this way will go towards the publication of a survey of humane computing.

Andrew Clement
789 W 18th Av
Vancouver BC Canada V5Z IWI

Humanistic needs some precision in definition, but it will be interesting to see what you find. Readers interested in working with Andrew are urged to correspond.

MORE COMMENTS ON COMPUTER TYRANTS

I am a student at San Bernardino Valley College and am majoring in computer engineering. Recently, I was leafing through some back issues of BYTE that had been given to me by a friend. I came across an article in the February 1976 issue by Ed Rush entitled "Could a Computer Take Over?" *[page 76]*.

In his article, Mr Rush points out all the speculation that various science fiction writers of today had written into their stories about the probabilities of a computer taking over some or all of future society. He suggested that all those who have any knowledge of the operation of an information processing system would be the first to confess this point. Not true.

I have only been working with computers for two years, but I have also been an avid science fiction fan. The idea of the computer as a world dictator makes me think of the naiveity of modern society. Those who do not want society to progress to an easier more controlled future have set the computer up on a pedestal as the ultimate enemy. Science fiction writers have taken this concern and have written upon it.

The true enemy is not the computer. The same people who tend to put down the computer are precisely the same people who

can appreciate its power. They are not afraid of the computer itself, but are afraid of those who control the computer.

The computer operators and programmers of the future will control the world, IF world power goes to the computer.

Richard G Castle
21951 Vivienda Av
Colton CA 92324

The neat thing, though, is so many different makes and models of computers doing most of the computing for individuals! Try to make that situation into a tyranny.

SUPER STAR TREK: TREK76

BYTE is fantastic and I'm now in the process of scrounging up enough money for a subscription. In your January 1976 issue there was a letter from Richard Wexler and he wanted info on STAR TREK programs. I have almost all the bugs out of an advanced game called TREK76 which should be just what he wants.

TREK76 is written in standard FORTRAN so it should run anywhere. Commands are entered in English sentences and the game is semi-real time; the Klingons attack and have the same armaments as the Federation. Once all the bugs are out of it (hopefully over the next few months) TREK76 will be one of the best STAR TREK games around. Anyone interested should contact me (school address is: Box 5739, University of Rochester, Rochester NY 14627 from September to May).

I'm a game freak so anyone with games to exchange just let me know. Also, I'm looking for a listing of a BASIC game called TREK73.

Dave Warker

SOME THOUGHTS ON "IGNORANCE IS BLISS"

The "Ignorance Is Bliss" television drive circuit (July 1976 BYTE, page 38) looks like a simple answer to a common need.

However, be sure to check whether the television circuitry is isolated from the power line. If not, use an isolation transformer for the television at any time that it is connected to the terminal. Otherwise, plugging in the television the wrong way will put 115 volts AC across the electrolytics if the logic supply is properly grounded, and will put 115 volts AC on the logic ground if it isn't connected to an external ground.

Also be sure that your television antenna has a good lightning arrester to take care of the voltage surges caused by nearby lightning strikes. This is especially important because the electrolytics cannot withstand as much voltage as the small, high voltage capacitors you could use with an RF modulator.

After you make certain that you are ready to use the terminal safely, you might consider that electrolytics aren't known for especially good high frequency characteristics. If you get a horizontal smear in the display, put a small (0.01 μF or so) bypass capacitor in parallel with each electrolytic.

Robert F Miles
242 Abingdon Rd
Lenoir NC 28645

Thanks for chipping away at "ignorance."

NEEDED: BENCHMARK COMPARISONS

One question which I have had for a long time concerning microcomputers is throughput. I have been using a PDP-8/E now for quite a while. Many of my programs (such as LIFE) were limited by speed. It seems to me that the 8 is considerably faster than many micros; but there are other things which I cannot adequately compare. For instance, there seems to be a lot more ways to address memory in a 8080 (or Z-80). How do these chips compare, in "benchmark" programs, to the minis?

How fast can the BASIC software for the 8080 count in a loop, such as I=1 to 20000? The PDP8/E, under the TSS timesharing system, managed it in about 20 to 40 seconds. Another computer I have used, the HARRIS Slash Four, did I=1 to 100000 in

about 15 seconds. At the same time it was supporting other users.

Simply counting makes for lousy benchmark programs, though. I would appreciate an article on the subject.

James Wiebe
208 N Jefferson
Hillsboro KS 67063

The benchmark slot has not been adequately filled with articles in BYTE, as James' letter points out. Authors take note: A general background article on the theory and validity of benchmarking, applied to microprocessor technologies, would be most useful to readers in evaluating processors.

DOCUMENTATION NEEDED

I am searching for a company that was at one time located in Derry, New Hampshire. The name of the company was/is Data Input Devices Inc.

They built a TV typewriter terminal unit that my friend (8800 Altair owner) bought at a hamfest.

We are hoping to get schematics from either the company (if it still exists) or from a prime consumer of the unit if we can find that out.

Any leads will be much appreciated.

Tom McMurtrey
WA4FYN
244 Roosevelt Av
Florence AL 35630

Can any readers help here?

A QUESTION OF COMPATIBILITY

Is it possible to successfully use the Altair 16 K static RAM (215 ns) in the IMSAI 8080 in lieu of the IMSAI 4 K RAM and still obtain the a) low power, b) memory (1 K blocks) write protect, c) power fail memory save benefits?

How about the MIKRA-D 16 K RAM in the IMSAI 8080?

I am seeking workable solutions to configuring a 64 K processing without using up (16 of 22) all the available slots while still getting the advantages offered by the processor manufactures on their smaller memory boards.

A B Clark
15425 SW 88th Av
Miami FL 33157

Has anyone in the BYTE readership tried this particular combination of modules? We have no data with which to give an answer to this question.

NEED AN ACCURATE HP-45 TIMER?

I have long been interested in the timing capabilities of the HP-45 and have watched for information on it. Last month *EDN* published how to add the crystal to make an accurate timebase, and said that HP was selling crystals for $14 plus $1 for shipping. Upon contacting HP I found they have them, but you have to know somebody. After some research and a few long distance phone calls I was able to make some arrangements to obtain a supply of crystals from the manufacturer. I would like to make these available to interested BYTE readers for $12 plus $1 shipping.

Enclosed is the article from *EDN* and also an explanation I received that differs slightly on how to install the crystal. This may be because of the two different versions of HP-45s. I have hooked one up for someone using the HP sheet and it works fine. The other way may work as well. I shall try it soon.

Mark S Egbert
1514 S 320 E
Orem UT 84057

PHOBIA

One of my pet phobias is people who short out capacitors. In BYTE #12, page 96, figure 1, the keyboard switches directly short out the 1 μF capacitor and this is, no doubt, the cause for the "oil film" referred to in the last paragraph on page 97. Just imagine the surge current that flows when a switch is closed. This is not good for the capacitor and is positively damaging to the switch.

Webb Simmons
1559 Alcala Pl
San Diego CA 92111

*Well, maybe. A capacitor of 1 μF, charged to 5 volts, holds Q = C * V = 5 * 10 ** –6 coulombs of charge. If the contact closure produces a spark, which can be detected on an AM radio, then it is a safe bet to say that the contact event takes place in from 10 to 100 microseconds (in order for significant harmonics to be detected in the 500 KC to 1500 KC AM radio band). Thus 5 microcoulombs are being discharged at an average rate of 0.5 ampere in the 10 microsecond case, 50 milliamperes in the 100 microsecond case. Peak values might be considerably higher, possibly pitting the contacts. This off the cuff analysis does not take into account the effect of the resistance in wires and capacitors, and is conditional on these contact conditions being able to generate measurable RF from a sparking contact.*

KUDOS, GRAPHICS, STATE OF THE ART

This letter of mine is in response to several items in your July letters column *[pages 90-94]*:

1. I would also recommend James Electronics (ICs). Delivery is usually less than a week.

2. With regard to the graphics display circuits:

   a. For the circuit you asked for I would recommend that consideration be given to making it 256 X 512, allowing it to simulate a standard 16 X 64 character terminal or display module.

   b. Any such circuit would not be inexpensive. It is not clear to me whether you are asking for a plug-in module, or a stand alone device to communicate through an IO port. Mr Hyde definitely wants a stand alone device. I doubt that a module would be noticeably cheaper than buying a Cromemco Dazzler (TM). The savings due to giving up color would be easily offset by the additional addressing, and they can get quantity discounts that the individual hobbyist cannot.

   A stand alone device would cost more yet: The electronics would be more complex, possibly justifying a dedicated microprocessor, and even worse, you then need a power supply, cables, a box and all the other things which so rapidly run up the cost of a good looking (and reliable) homebrew project.

3. I am afraid that the hobbyist is never going to get noticeable quantities of true state of the art items. Announcements in magazines such as *Electronics* are at least six months (if everything goes well, which it usually doesn't) ahead of volume deliveries. After that, there is at least another six-month wait to design an item into a hobbyist kit, and get that into production. The probability approches 1 that one of these steps will meet an unforeseen delay of at least 90 days, such as the capacitor shortage of a few years ago.

Finally, the cost of truly new items (4 K RAMs, not cheaper floppies) is far more than the hobbyist can afford. 1 K RAMs cost a darn sight more in thousand quantity five years ago than we are currently paying for one. I worked with 93410s (256 bits, 50 nsec) when they were $20 apiece in hundreds, and they're now $1.95 in onesies.

As this happens to 4 Ks we'll start seeing them.

Anyway, congratulations on a worthwhile and continually improving magazine.

Frank Richards
414 Bradford Way
Norristown PA 19401

AUTHORS, TAKE NOTE

Ever since BYTE #1, I've been waiting for more conceptual articles like "Write Your Own Assembler." How about some more articles on . . .

- Random number generators, and the testing and demonstration of randomness in various random number generation schemes.
- Executive software structures, what kinds exist, advantages and disadvantages of various methods, etc.
- Human engineering factors and pointers concerning computer peripheral placement, panel and console design, etc.

James F Gentry

NEEDED: USED COMPUTER INFO

I am a charter subscriber to BYTE. I have enjoyed every issue from #1 to date. Your publication has expanded greatly my knowledge of software. Mostly I have learned that I have much to learn.

Like everyone who has been bitten by the computer bug, I wanted very much to have a system to play with. I had almost decided to embark on the purchase and assembly of either a 6800, or an 8080 system kit (with faint echos and moans from the wife), when I chanced upon the purchase of a full blown 16 bit computer through a local government surplus outlet. The only problem with this was that some of the interconnect cables

were missing and there was NO INFORMATION.

The unit I have is a DATA 620 system, housed in two 19 inch wide, 6 foot tall racks. It consists of a CPU, two power supplies, a bootstrap loader panel, tape IO and reader (used to load the program, I believe), a general purpose IO (purpose unknown), and two core memory units (organized as 2048 words of 16 bits each unit, 4 K total CORE).

This unit was in use at the Manned Spacecraft Center in Houston (use unknown). I believe it was in use circa 1961. It was probably built around that time because it uses all discrete logic, no integrated circuits. It uses −10 volt load source, +6 volt logic bias. The machine was made by a company called Data Machines Inc, Newport Beach CA. This company is/was a subsidiary of a company called Decision Controls Inc (address unknown). As you can see I have more questions than data. Although a chance investment, I thought it a good buy.

It is my deep hope that you or one of your readers might be able to supply me with some information. I have written to Newport Beach c/o the postmaster, but no such company is in existence out there, and no forwarding address. Any help you can give me in this matter would be greatly appreciated.

Godfrey C Leggett
185 Marceline
Beaumont TX 77707

We publish this letter in the hope that some reader will provide information helpful to Mr Leggett in his pursuit of a personal system.

VA COMPUTERS NEEDED

I have recently become aware of the world of personal computing and am anxious to become part of it. Like many others, however, I am not prepared to lay out the money for a microprocessor, IO device, memory, permanent storage capability, etc. If there were microcomputer kits, as there are television kits, which were part of a Veterans Administration approved correspondence course, the door would be open for thousands more. Could you put me in touch with anyone having such a course or spread the word to the microcomputer kit industry about this seemingly untapped, eager pool of potential customers? Thanx.

Steve Packard
44 S Greenfield Av
Hampton, VA 23666

IBM SELECTRIC IO

The letter from Dr James Lang in the August BYTE on the desirability of using the IBM typewriter for a console device hit close to home – after I design the interface for this Selectric 735 I'm typing on, that's what I will have. Please note – this isn't just any old Selectric, but the one with all the solenoids and switches already under it. These machines are now available on the

Continued on page 75


Sol Terminal Computer
Processor Technology

Continued from page 71

used market for prices ranging from $500 to $1000, depending on model and condition. This machine cost $1 K, as it was a new machine bought from one of the used mini houses. I am working on an interface to connect this gadget to my IMSAI 8080.

This interface will be compatible with most 8080 software; that is, the code will be ASCII-7, with port selection and flag bits jumper-selectable. I also intend to make the interface compatible with most, if not all, of the available options for the 735 and 731. (The 731 is the narrow version of the 735, with an 11 inch writing line instead of 15 inches.) I have heard that the 735 has been replaced in the IBM lineup by something else — this may be the reason they are showing up in the used computer shops. Whatever the reason, it's nice to be able to get good machines at reasonable prices.

Philip M Spray
3006 N E 26th
Amarillo TX 79107

Avoid mechanical kluges — buy one with the actuators built in.

ARTICLES NEEDED: GETTING SENSOR DATA INTO COMPUTERS

I have really enjoyed BYTE, especially articles such as the Suding and Lerseth contributions in the July issue. I wonder if you will be doing some articles on connection to the analog world. This seems to be an important region that has been shortchanged in your publication.

I would like to see transducer data such as thermocouples, thermisters, and RTDs for temperature measurements; strain gauge data for stress, pressure, etc; optical and/or magnetic transducers for velocity, etc. I would also be interested in explanations about signal conditioning (bridges, channel amplifiers, zero set and scaling amplifiers), A to D and D to A techniques and methods (such as ramp DAC, successive approximation, and tracking A to Ds). For a micro, or any other computer, to "feel" what is going on in the physical world, these sensors that collect ignition pulses, temperatures, load pressure data, etc, are very important.

C Southard
2519 Meadowbrook Dr SE
Cedar Rapids IA 52403

TIMMMBBERRR!

Enclosed you will find my check and subscription to BYTE. I've only been able to find a few copies in this area and I found each of them very interesting and informative. Having just gotten our Altair 8800 up n running I thought you and your readers might be interested in hearing about its unique application. Presently, we are using an 8800 at this Lumber Mill to monitor our daily production. We have a lumber sorting machine that incorporates CMOS technology in its control system. Through the use of limit switches the machine measures every board as it sorts it coming out of the sawmill. It displays the measurement data on a set of seven segment LEDs built into the control panel. That is where we tapped in our interface, designed for us by Warren

Stardup, a partner-technician with Micro-Computer Systems Inc, which is based in Tampa FL.

The interface converts the TTL at the display to serial and relays the data via 20 mA loop back to the office where the computer is located. Currently, we are using a video display with data in memory being *dumped* on cassette every hour, in case of power failure and loss of memory. A hard copy printer is on order. In the future I plan to scan the logs just before they are cut, compute that versus what comes out at the sorter, thereby giving me a readout of the recovery I get out of each log. Thought this might be of interest to you. Keep up the good work. Your magazine is refreshing!

Tommy Staten
HC Hodges Lumber Co
Panama City FL 32401

THE EMULATION BLUES

I read with interest Don Keek's letter in the August issue of BYTE concerning the emulation of a PDP-8 instruction set on one of the popular microcomputers. As the author of several emulation programs for various computers, I feel obliged to comment on my experiences. I have written two emulators for the PDP-8, one implemented on the Control Data 6400, a 60 bit scientific computer, the other on the PDP-8 itself. The self emulation has value in a trace-type debugger for programs and peripheral interfaces. The CDC 6400 implemented emulator was used to teach minicomputer programming techniques to a large number of students in a batch environment.

As Don mentioned in his letter, memory and execution time are definite bottlenecks in emulating one machine on another, particularly when the characteristics of the two machines are different. I would suggest that a PDP-8 emulator program written on an 8 bit micro would be hard put to achieve average execution times of less than 20 times the execution time on the PDP-8 itself. Memory requirements would be about 10 K bytes for emulation of a 4 K PDP-8.

The higher level languages supported on 4 K PDP-8s are almost all interpreters (FOCAL, BASIC, and 4 K FORTRAN for sure). The further degradation of an interpretive language by at least a factor of 20 would be quite noticeable, if not annoying. *[Total degradation: 400 times slower.]*

Quite a few PDP-8 programmers have made use of "programmed delays"; that is, looping a specified number of times to achieve a specific time delay, or to determine whether a peripheral has stopped sending data. Such loops will obviously not emulate faithfully.

I recommend that anyone interested in writing an emulator acquire the CPU diagnostic programs for the machine being emulated. This will be an invaluable aid to debugging and benchmarking.

R A Schottland
7040 N Sheridan Rd
Chicago IL 60626

DIVIDE AND CONQUER (INSTABILITY)

"Why Wait?" by Suding *[July 1976 BYTE, page 46]* is very interesting. The ratio of his two frequencies is 1 to 1.4. Many of us do not have access to counters or oscillators. Crystals and common chips are cheap.

If a 1 MHz crystal oscillator is divided by 32, the result is 31250 Hz. If this is sent to a divide by 14, and a divide by 10 chain, then the results are 3125 and 2232 Hz. Switch between the two with an inverter driven by the data, gate it into an integrator, and one gets the triangular wave to drive the tape head. This should eliminate stability problems in the two frequencies and should still be within the telephone line frequencies for acoustic couplers for phone transmission. Many of us would like to be able to use cheap cassette recorders if possible. Suding implies that his filters can be tuned over at least one octave, and maybe two. If this is true, R25 and R26 should be reduced in value and padded with resistors.

However, if considerable variation is possible in frequency, and a long period of each tone is put on the tape before the program, say 30 seconds each, and a voltmeter were connected to IC37 pin 7, then the filters could be tuned (max volts out) to the frequencies actually present, and the only requirement on the cassette is that the speed be within plus or minus five percent for the time the program is running.

I regret that I do not have the equipment to try this system, but I am willing to discuss it with anyone.

Cecil H Royce
255 S Marion
Oak Park IL 60302

The point about 3125 Hz being compatible with the phone network may be debatable; would any telephone engineers care to elaborate? There is a standard set of modem frequencies available, in integrated circuits yet.

A point about standards which should be repeated: The way to approach the subject is to document what is being done by manufacturers, with discussions about the virtues (or faults) of the systems. This way, the user (and other manufacturers) are kept aware of the state of the art as practiced in manufacturable systems. So far, there are two tape interface definitions documented in BYTE for audio use; more will come.

SOME THOUGHTS ON MISCELLANY AND TAPE MODULATIONS

I just finished reading the August issue of BYTE pretty much cover to cover and as usual found it interesting. I wanted to write last month but didn't get around to it and I have a few tid Bits (BYTES) of information and complaints to throw out.

One minor bug in BYTE is I wish you guys would put the issue date at the bottom of each page. It makes it rather hard to reference some of the older articles that I Xeroxed from a friend's earlier issues.

I thoroughly enjoyed "Jack and the Machine Jive" by Bob Grappel. It's been a while and I honestly forgot how hashing works. It was a very interesting and informative refresher course. I am presently giving some thought to a "general purpose assembler," one that can be used to assemble code for more than one micro-widget simply by changing the lookup tables.

Martin Buchanan's Video Disk article and Burt Hashizume's writeup on the Zilog Z80 were also very interesting. The Z80 will most certainly become the most popular micro as soon as the price drops to the same magnitude of power as the 8080 and 6800. I really like the maskable and unmaskable interrupts and arithmetic shifts features of the 6800 and Z80. Does anyone know an easy way to do an arithmetic shift right using the 8080? But Zilog has outdone themselves with the bit addressing, block transfer (flush instruction) and search instructions. The mode 2 interrupt is also the way I prefer to do vectored interrupts, but there are also two other ways (modes) if you are fussy.

John Baird's comments on the information revolution are beautiful. We all know that BYTE can do what no technical trade journal can do: Convey practical, down to earth, State of the Art, easy to understand ideas. The technical journals have mostly become a place for doctorate fellows to shelter a highly technical snow job just to meet their degree requirements. Decipher and remove the high level math and the guy usually has little information to pass on. BYTE is the source of innovative design ideas (hardware and most certainly software).

I was a little upset at Dr Suding's "Why Wait? Build a Fast Cassette Interface" article in the July issue. His article would lead you to believe that frequency shift keying (FSK) is the only way to fly. Of course any article written by an author writing about a product he is trying to sell is usually a little biased. Dr Suding is obviously an analog man with his filters and tune up notes. Tweek this . . . Tweek that. Sure Manchester employs "harmonically related frequencies," and frequency discrimination techniques just won't hack it. He has one good point: "Imagine reading 300 baud for 15 minutes to discover a noise pulse had destroyed data, requiring re-recording." This problem exists no matter what recording technique is used. The best solution is to use short data records; say 64 bytes (also a good format for listing). Then if your cassette has backspace capability you simply back up one record and try again instead of a complete rewind. Using character, record and file gaps is a worth while consideration for a well managed file system. Try to keep the record size fixed to whatever corresponds to your single line list device, at least the source statements. A file is a collection of records, either a program or subroutine.

Back to FSK versus PSK: First of all Manchester is really phase shift keying (PSK) using square waves instead of sinusoids. Both FSK and PSK require an extra start bit or sync bit equal to a one. Even using the Suding FSK there is no physical way to go from DC (no flux changes at the start of a cassette record) to a 2975 Hz signal, a 0 without first going through 2125 Hz, a 1. Dr Suding needs to review his Fourier transforms if he believes that only 2125 and 2975 Hz frequencies exist in the FSK waveforms. The PSK (Manchester or similar) method requires the first bit sent to be defined arbitrarily as a 1. From thereafter one definition of Manchester may be a zero phase shift meaning no change in the state of the data (ie: a 11) and a 180 degree phase change in the signal means complement the

data state (ie: 10). Another definition is a low to high transition in the center of a bit cell is a 1 and a high to low transition at the center of a bit cell is a 0. In any case the Manchester code consists of two pulse widths: full bit cell and half a bit cell width. A digital demodulator needs then to decriminate a 1/2 bit cell pulse from a 1 bit cell wide pulse. This is usually done with one or more delay elements (one shot or delay line) set at a 3/4 bit cell reference point. The Manchester or similar codes potentially have a faster data rate than FSK.

For example, say the upper recording limit on a cassette was 2 kHz. The Manchester data would be recorded at a 1000 bits per second rate. The two pulse widths of 1 msec (half bit cell) and 2 msec (full bit cell) would be involved. One bit cell only requires 1/2-1 kHz cycle or 1-2 kHz cycle per bit cell. The FSK technique using filters I doubt seriously would be able to respond to a 1-2 kHz cycle pulse. It would certainly require a bunch of 2 kHz cycles (the modulating frequency is obviously got to be less than the carrier). Even if it could detect a 2 kHz signal in 4 cycles, say, then Manchester beats the FSK 4 to 1 in recording density (speed).

Also since there is always at least one flux change per bit cell using Manchester or similar self clocking code, the demodulator can correct for variations in tape speed by adjusting its 3/4 cell reference either using a phase locked loop (PPL) (see "Improved Cassette Interface Circuit" by Hal Mauch, page 8, April 1976 BYTE) or a programmable counter or even a software tracking routine.

A software routine could count the number of elapsed clocks since the last flux change (phase detector) and decide if it was a 1/2 cell or full cell change. It could take this number, subtract it from what it was expecting for a 1/2 or full cell clock count, then average the error over previous samples (low pass filter) and make a correction (prediction actually) to what the next 3/4 cell reference should be. The routine would be realizing what is called a sampled phase locked loop. The only readback hardware need only be a limiter similar to Dr Suding's. The rest could be software. If a data transmission is going on, the micro is probably not doing anything but waiting around for the character to shift out anyway.

Frank Bennett
Hughes Aircraft Co
Displays Lab
Fullerton CA 92634

One way around the tape interface problem is the obvious one, a point raised in your letter which we emphasize here: With the proper general purpose signal conditioning hardware to convert AC coupled signals into TTL digital levels and vice versa, nearly any modulation method can be generated and decoded using simple algorithms. Several computers currently being marketed use such generalized hardware backed up by software in PROM monitors. These include the new Apple computer, BABY by STM Systems, etc.

SOFTWARE

People want software. There are about 1,000 computer programs in the public domain. They are available to anyone — including hobbyists — at cost. These programs include all of the common computer languages. They obey the instructions of most terminals.

There is no need to reinvent the wheel — again, and again, and again.

The above programs may be readily translated into 8080, 6800, F-8 6502 microprocessor instructions. There are no restrictions on revising these programs. There are no restrictions on distributing them. They are in the public domain.

Why don't you do yourself, your customers, your subscribers, and thousands of others a grand favor, and write for information from

Documents Office
Kiewit Computation Center
Dartmouth College
Hanover NH 03755

Edward L Tottle
2993 Yorkway
Baltimore MD 21222

HOW TO WIN AN ELECTION

As a professional software engineer (a fancy term for a programmer), I receive a great many computer related publications each month. I am six months behind in reading *Communications of the ACM*, two issues behind of *Computing Surveys*, four back of *Computer Design*; and I don't even want to think of how far of *Scientific American*. But every month, as soon as it appears in the mailbox, BYTE gets read from cover to cover.

In the free market, there is one sure way to show one's appreciation for a product. Enclosed is my renewal for three more years of BYTE. Keep up the good work.

Howard L Turetzky
HYDRA
1575 Ivanhoe St
Denver CO 80220

PS — This letter was typed on an IBM Selectric IO writer driven by an Altair 8800, so blame all the typos on the computer.

OLDE ENGLISHE

I have been on a crusade over the past few years, trying to discover the true origin of the word "byte," but my efforts have been unsuccessful. I have begun to think that the origin is lost, but I have decided on one last attempt.

Since the name of your magazine happens to be the very same word that has been the source of my frustration, I am hoping that you can shine some light on the origin of this small word. I have looked in just about every textbook that I can get my hands on.

If you can provide some relief to my plight, it would be most appreciated. Thank you for your trouble.

Thomas P Bishop
PL/C Project Manager
Department of Computer Science
Cornell University
Ithaca NY 14853

*I first ran across an etymology of the word BYTE in a book I read on the IBM "Stretch" computer borrowed from an associate of mine at Intermetrics Inc, where I worked prior to founding BYTE. In that book, which was published in the early 60s, a research precursor of the IBM/360 series was being described. The "Stretch" computer (if I remember correctly) had a large bit addressable memory, in which the term byte was an arbitrarily coined word used to reference an arbitrary bit string field of length "n". The term at that time was meant as a generalized concept of a bit string subfield. When the 360 came out, all that changed, since after System/360, almost all published literature references 8 bit bytes. Perhaps a reader can supply a more definitive answer to the question of the term BYTE's origin, and some of the history of the early "supercomputer" work following the "second generation" transistorized computers of the late 1950s and early 1960s. . . .***CH**

FLEXOWRITER FLEXIBILITY

4891 Songbird Dr.
Columbus, Ohio 43229
31 July 1976

Dear Sirs,

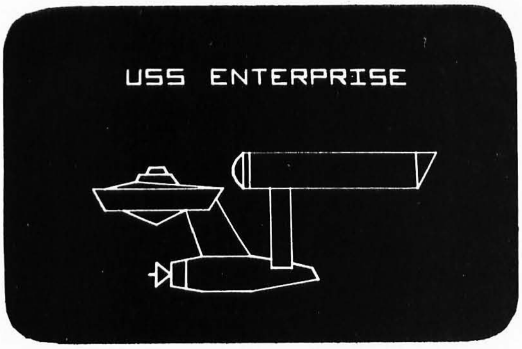
I see a number of complaints about the lack of reasonable priced IO devices available which will produce hard copy. This letter is being typed on a Friden Flexwriter I purchased for $200 used. There are many available for less. As you can see it has an attractive typ face an both upper and lower case,, It has a paper tape punch and reader., but a very poor typist. It runs about 10 cps. Although it is not connecter to a computer at the present it has connections for data input. I(rather my sec) uses it for a lot of red letter writing. I am sure someone could work out the problems of connecting it up. The paper tape is a seven track tape.

Sincerely,

Lawson F. Pierce

Lawson F. Pierce

Sounds like an excellent hard copy deal.



Make Your Next Peripheral a Real Eye Opener

Steve Ciarcia
124 Hebron Av
Glastonbury CT 06033

Photos by Ed Russo

Have you ever gone to a party and been entertained with some party game which the host introduced to impress his guests and salvage an otherwise boring evening? American life is built upon a succession of gimmicks. We feel we have to be entertained to exist and cannot just have a good, dull time. An example is the success of television as the universal babysitter and now all these overly expensive television games.

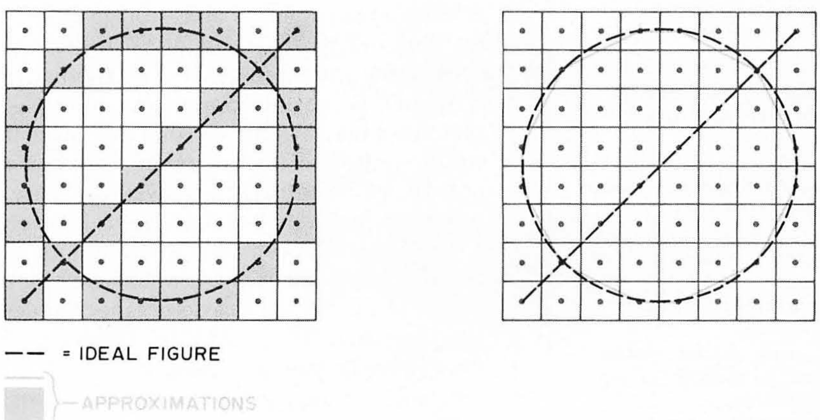
Excuse my "Holier Than Thou" attitude, but I'm no different than the rest of the population, including having one of those stupid TV games. What I'm trying to lead up to is to introduce a microprocessor driven vector graphics display controller which can be built for $60 or less.

Some history is in order. About eight years ago, I was at MIT for a conference and spent some time touring the campus. In the corner of one room in an otherwise deserted computer center was a large, intensely involved and vociferous crowd. Above the usual crowd noise were heard shouts of "Shoot your phasers now!", "Watch out for that meteorite!", and "He's in the space warp!". Because I usually avoid crowds because of something my grandmother once said about the correlation between crowds, fleas, and bubonic plague, I fought off the urge to investigate further. But, after hearing, "His antimatter engines are done — blast him out of the sky!", and tumultuous applause, my curiosity could take no more. It took about five minutes of pushing and shoving to make it through the 15 deep crowd to get close enough to see what was going on. There were two people sitting in aircraft type chairs holding what appeared to be small control consoles. They and the entire crowd were intently watching a 21

Figure 1: Raster Scan versus Analog Vector Graphics. For a given grid size and a figure covering a particular area (such as 4 by 4 or 8 by 8), the analog vector graphics method will give a better picture, where better is defined as a closer approximation of the figure desired.

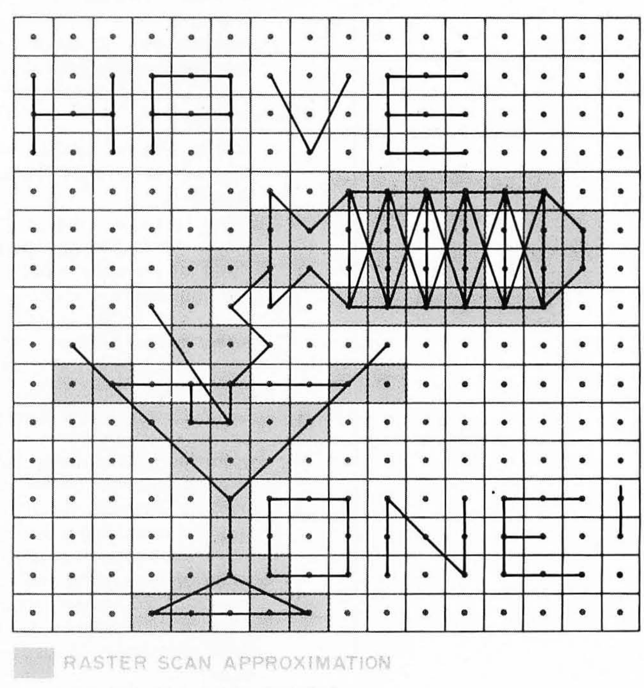


a. Drawing a circle with a slant line through it is unrecognizable on a 4 by 4 grid of raster dots, but quite believable when drawn on a 4 by 4 vector graphics grid. A line drawing of the "ideal" circle with slash overlays the approximations for reference.



b. Drawing a circle with a slant line through it is a closer approximation on an 8 by 8 grid of raster splotches, but the 8 by 8 equivalent vector line drawing has a still closer approximation to the desired circle and line. A line drawing of the "ideal" circle with slash overlays the approximations for reference.

c. When it comes to detailed graphics, the vector method is a real winner. Here is a cartoon figure rendered on a 16 by 16 grid in both vector and raster scan dot matrix form. The contrast in effective resolution of the two methods is enhanced by the juxtaposition of the two versions in this overlay. Note that within the same coordinate grid, the vector graphic version is able to represent some limited alphanumerics, impossible to even attempt in the raster graphics version.



inch black and white TV screen upon which there was a scene depicting a star-studded space and two rocket ships. It became obvious very soon that they were playing "Space War," and the only rules were to shoot the other guy out of the sky (Do Unto Others. . .), while not getting wiped out yourself. There were certain limitations on fuel and armaments which made the game all the more interesting.

I stood and watched this for three hours until I jockeyed myself into position to grab a control console from an exiting player. All right, you guys, let's play Space War! I played for what seemed to be an instant on the time continuum; but it was obviously too long in the crowd's opinion because, when some little kid started pulling one of my socks down and I let go of the control console to retaliate, I was six deep in the crowd again before I regained control. I did eventually get another try, but I was already a confirmed "Space War" freak. I was determined to have one of my very own that I could play until my heart was content.

The realization of this dream was, of course, to be postponed until just recently. It is now possible for the computer experimenter to put together a system with capabilities rivaling computers costing many times more. The LSI microprocessors which have become increasingly available at modest prices have started a boom in the home computer field with price wars inevitable. The major reason that these small computers have realized so much computer power is that designers have tailored their designs to utilize existing hardware which is available readily. Consider mass storage, for example. Talk to IBM or any "traditional" computer company about mass storage and your discussion will lead to a $30 K tape drive and controller with reams of wonderful tolerances and grandiose specifications. This will provide an unparalleled mass storage capability, but most people would be upset just considering the electric bill. The alternative was, and is, to incorporate the principles that make the "biggies" so good, but make it cheap enough so the average person would not go into cardiac arrest over the price. American ingenuity had the answer. Develop a sophisticated, yet simple modem on a cheap printed circuit board which allows the owner to use his kid's cassette tape recorder to store data. Thus came the invention of the $100 mass storage system.

There is a moral to all this. The initial success of the home computer system can, in large part, be directly attributed to inexpensive peripherals which utilize readily available surplus equipment or home entertainment devices such as tape recorders and televisions. The television typewriter, otherwise called an alphanumeric CRT display, is only $150 because it uses an already existing television.

RASTER SCAN GRAPHICS – SUMMARY OF ADVANTAGES AND DISADVANTAGES

Advantages

1. *No software refresh* – computer can load a display once and continue other duties.
2. *Fully digital* – does not require any analog conversion or associated "tweeking" of components.
3. *Requires no special display tube* – uses television monitor.
4. *Can be color* – by using added storage, color or gray tones can be assigned to each element or group of elements.
5. *No limit to total display quantity* – since storage is provided for all points it is irrelevant whether they are shaded or not and does not affect display refresh time.
6. *Inexpensive* – in limited forms, raster scan uses inexpensive televisions or monitors.

Disadvantages

1. *Not a pure line display* – display is a connection of shaded squares and not line segments.
2. *Must provide memory for total display* – for a 256 by 256 point display, 8 K bytes of memory must be provided even if only one element is being displayed.
3. *Must use either an external memory or direct memory access* – the composite video signal necessary for raster scan requires a data output rate beyond the capability of the processor itself. Special controllers must be provided to either access standard program memory directly by DMA or scan an external display buffer which can be periodically changed by the processor to correspond to changing displays.
4. *May require fast memory access* – in DMA using cycle stealing, memories having access speeds of 250 ns or better may be required. (With a dedicated buffer memory or multiport memories, 2 μs access is typically required.)

Raster Scan Graphics

There is an unfortunate lack of sophisticated peripheral devices beyond these few, and applications using these are limited. The Space War game previously mentioned requires a graphics CRT terminal upon which the players and playing field may be displayed. The home raster scan television, such as that borrowed for an alphanumeric CRT, can be used; but the display is overly complicated and will appear as a connection of blocks rather than pure line segments. There are many raster scan graphics systems now on the market but all exhibit this "block" phenomenon, which in the simplest terms is a function of the resolution. A low

resolution display will have a limited number of large blocks and a high resolution display will have a great quantity of small blocks. Making various blocks light or dark is what constitutes a picture. Figure 1 illustrates a circle drawn on a screen with 16 elements (4 by 4) versus one which has 64 (8 by 8) elements. The lower the resolution of the display medium, the less recognizable the display. For true graphics detail necessary in applications such as game boards and players, high resolution on the order of 256 by 256 is required. This leads to another interesting problem: display storage and addressing.

Raster scan graphics is nothing more than a sequential video display of mass memory. The "mass" in mass memory is a function of the resolution. If the display has a total of 64 elements, 8 bytes (8 bits each) of memory are required for storage. If the display were 64 by 64, there would be 4096 picture elements requiring 512 bytes of memory. It is important to remember that the entire 512 bytes is necessary regardless of the picture being displayed. Within this display storage, each bit has to have a unique X and Y address: For a 64 bit display X and Y could each be defined with 3 bits; for a 4096 bit display 6 bit X and Y addresses are necessary.

Most home computers are 8 bit machines and obviously work most efficiently with 8 bit manipulations. Taking full advantage of this and defining 8 bit X and Y display addresses results in a very respectable 256 X by 256 Y display grid containing 65,536 elements. This corresponds to 8 K 8 bit bytes of memory. *[See the article by Thomas R Buschbach on page 32 of this issue for an example of such a design.]*

Going from the memory to the screen is another problem and requires a special controller to translate the memory into video information. Horizontal scanning rate of a television is 15.75 kHz which is 63 μs. If a horizontal line is to be divided up into 256 elements, each must be scanned and output to the TV one every 250 ns. Even if addressing and output were handled 8 bits at a time, the rate will still be one byte every 2 μs. Very few microprocessors are capable of doing this. There are two methods of acquiring this data. First is to have a dedicated 8 K memory in the graphics controller which is synchronously scanned by the video output generator and asynchronously updated via the output ports of the computer. This method, though easiest from a design perspective, is the most costly due to the extensive memory used only for the graphics controller.

ANALOG GRAPHICS – ADVANTAGES AND DISADVANTAGES

Advantages

1. *Does not require extensive memory* – total memory required is whatever is necessary to store all the beginning and end points of the line segments to be displayed. Blank areas do not require storage.
2. *High resolution and low cost* – a simple controller is designed around low cost DACs and an analog display does not exhibit "staircase effect" on diagonal lines.
3. *Uses available test instrument as display* – an inexpensive oscilloscope or other X,Y display may be used as a display medium.
4. *Does not require fast memory* – any storage medium having a byte access time of approximately 100 μs is adequate. Can use shift registers, delay lines, etc.
5. *Resource conservation* – gives the author some reason to use his 8008 system since he bought an 8080A.

Disadvantages

1. *Software refresh* – the processor must periodically rewrite data on the screen to keep it from fading. Refresh rate is a function of processor speed and the total number of line segments in the display.
2. *Uses analog circuitry* – all analog circuitry is subject to drift and to temperature and other external influences. These variations can be minimized through proper design and construction.

The second method which is used most often in the graphics industry is called direct memory access. 8 K of memory is still required for the 64 K element display, but it is attached to the computer. It is directly addressable at all times and not externally located in the graphics controller, so that when the graphics option is not in use, this 8 K is utilized as any other portion of memory. When using the graphics operation this 8 K will contain the bit pattern display of the screen and be shared between the processor and the graphics controller. Direct memory access means just what it says. The DMA controller directly addresses and reads memory when the processor is not doing so. In most microprocessors this sharing can be done either by suspending the processor operation with hardware DMA request and hold logic or by accessing memory on clock cycles when the processor is doing something else. Suspending processor operation is easiest from both a design and component standpoint but least effective when simultaneously running large programs. In a raster scan display, the processor would be shut down and memory read out 30 times a second to correspond to the non-interlaced update of standard television. The only time the processor would be allowed to run

would be during vertical retrace which is approximately 4 ms out of every 33 ms. Therefore about 400 instructions (in an 8080A) could be run every frame. This is quite acceptable for most non-iterative programs, but it would kill response in interactive Space War type games which do a fair amount of number crunching. Modifications to this method can extend processor time somewhat but they become involved. DMA utilizing "stolen" clock cycles or added wait cycles is the most effective technique since it does not significantly reduce processor calculation speed. Not all microprocessors are capable of doing this, though; and many more design constraints are involved. Another method which avoids waiting by the processor is to use a multiport memory with priority allotted to the processor. This leads to some picture breakup during updates but allows the processor to run at full speed.

Some such form of direct memory access is the much preferred system for high resolution raster scan graphics but can be very expensive when considering the quantity of memory necessary for display storage.

Analog Graphics

For those of you that haven't ordered a raster scan graphics controller after reading the previous dissertation, I would like to present the other side of the story: analog graphics. Usually the thought of anything analog associated with a digital computer is abhorrent to the purist. Remember, before there was digital there was analog; and even though it's hidden, analog's still there. The success of digital systems is based upon their ability to evaluate an action or output a value repeatably with no deviation from previous occasions, given constant input data. This is inherent in the machine structure, since it can only exist in either of two discrete states: 1 or 0, on or off, etc. Analog on the other hand can be influenced by external conditions such as temperature and power supply levels, etc, and is repeatable only within a more or less closely defined tolerance band. Industry has been tightening this tolerance band in recent years to such a point that analog circuitry becomes too expensive and digital methods are implemented. Too often, though, the "band wagon syndrome" prevails. Many applications which can be adequately accomplished using analog techniques have been converted to digital. In most cases this unnecessary conversion to digital can be very costly and requires expensive support equipment. The tolerance tightening which makes an industrial manufacturer use only digital does not exist for the home computer enthusiast. Many applications can be more cheaply done with analog. A prime example of this is graphics.

Analog graphics is a method which converts digital data stored in memory to voltage levels proportional to that value. Two of these digital to analog converters would be incorporated to provide two simultaneous outputs which represent physical X and Y coordinate locations on a display screen. The display's X and Y full scale deflection would be directly proportional to some analog constant times the digital value in the memory location being accessed. The display medium is an oscilloscope or similar two dimensional proportionally deflected display.

As was the case with raster scan graphics, analog graphics has both its advantages and disadvantages. The most pronounced advantage is the display method itself. Two bytes of memory converted to analog X and Y positions define the starting point of a line segment and two more set the analog values of the end point of that segment. When triggered by the controller a line will be drawn between these points. Four bytes define the entire line regardless of the length (within the limits of the screen dimensions of course). There is no need to define and store points along the line, only the beginning and end point. When using an 8 bit X and Y addressing system there are 64 K total locations between which line segments may be drawn. The path between any two points is direct line and need not coincide with an addressable point location. This results in an extremely high resolution display even though it is only 256 by 256. This means that in a diagonal line there is absolutely no "staircase effect" inherent in a raster scan display. This and relative low cost are the chief advantages.

The major disadvantage of this system is that it requires software refresh. When beginning and end points are sent to the controller a line is drawn – once! It will be seen briefly on the screen and quickly fade unless it is refreshed. The computer must do whatever it has to do and get back to the graphics driver program and write the same line segment again before it fades. This time between refreshes is a function of phosphor persistence, a characteristic of the CRT being used. 25 to 40 ms is usually the maximum and if exceeded will result in display flicker. (Storage tubes exist with longer time constants, but at much higher expense.) The number of lines which can be written and refreshed is a function of processor speed. The faster the processor, the

sooner it will be able to write a screen, do its own thing somewhere else, and return to start writing the screen again. All the displays in this article were driven by a slow 8008 processor (not an 8008-1) system and about 120 lines could be drawn before any flicker was perceptible. This was using a worst case programming technique (the one I'm best at) which I'll explain later. Faster processors would of course increase the number of lines capable of being displayed, but would have to have some delay loops in the software so as not to overdrive the graphics controller. The digital to analog conversion does take a little time.

An Analog Graphics Controller

The circuit for such an analog graphics controller is illustrated in figure 2. This circuit is based upon a design originally proposed by Hal Chamberlin in the first three issues of *The Computer Hobbyist*. Modifications and additions have been incorporated to allow more ease in calibration. The circuit consists of three major sections: digital input latches, digital to analog converters, and vector generators.

Since the graphics controller is software driven, possibly the software side is the easiest starting point. Positioning and drawing a line on the display is accomplished through four output commands from the processor. These commands are not some new processor instructions but rather decoded output strobes which transfer 8 bit position data from the accumulator to the controller's input latches IC1 to IC3. These four strobe commands are appropriately referred to as X-MOVE, Y-MOVE, X-STORE, Y-DRAW. In actuality, X-MOVE and Y-MOVE are the two bytes of data which define the starting location of a line segment while X-STORE and Y-DRAW is the end point of that line segment. There are four separate outputs necessary because we assume an 8 bit machine and 16 bits of XY positional data is needed to define a point on the screen. Obviously, in a 16 bit machine, this process could be reduced to two output instructions. Extra decoding is often not necessary. Many microprocessor systems like the Scelbi series provide extra decoding logic on the back plane wiring while not mentioning it. The usual clue is to check the IO decoder board in your system and look for an appropriate decoder chip such as a 7442 or 74154. Often only a couple of output strobes are utilized on the chip. In the case with the 7442 used in the Scelbi, the extra strobes allow quick unaggravated hookup to the four necessary for graphics.

IC4 and IC5 are 8 bit digital to analog converters. Their function is to convert the 8 bit position data into a current which is proportional to that value. IC6 and IC12 are current to voltage converters. The Motorola MC 1408-L8 consists of a reference current amplifier, an R-2R ladder, and eight high speed current switches. A resistor between a reference voltage and pin 14 is set to produce a reference current of approximately 2 mA. The R-2R ladder divides the reference amplifier current into binary weighted components which are fed to the switches. The output current from pin 4 will be proportional to the binary input to the converter. For example, if the binary input were equivalent to 128 decimal, then the output current would be (128/256) * 2.0 mA which equals 1.0 mA. The current to voltage converter is set to produce −2.5 V for a 00 000 000 binary input and +2.5 V for 11 111 111. Exact calibration procedure will be given later.

The heart of an analog graphics controller is the vector generator. This circuit does the actual drawing of the line on the display. Merely setting a starting location and jumping to an end point will not work. The beam's transition between the points would be so fast as to leave only a very weak trace if any at all. A vector generator provides a consistent drawing time of approximately 100 μs, allowing for a very bright trace.

Actually drawing a line segment is a simple series of events. First, the processor outputs sequentially the X and Y starting position coordinates with the X-MOVE and Y-MOVE commands. These values are stored in IC3 and IC1, respectively, and cause the digital to analog converters IC5 and IC4 to follow suit. Within 3 μs, IC6 and IC12 have settled out to a voltage representative of these coordinates. IC10 and IC11 are analog switches which route the raw X and Y voltages through the vector generator. These CMOS switches, equivalent to SPST, are in the off state when their control terminals are at a potential equal to V_{DD} (−7.5 V) and on when it is at V_{SS} (+7.5 V). IC19 is a voltage translator which converts TTL switching levels to those necessary for this CMOS device. An alternate circuit to replace the DM8800 is also provided.

In the quiescent state switches SW1 and SW3 are on and switch SW2 is off, as controlled by signals T1 and T2. This allows the output of the vector generator to be equal to the coordinate position of the beginning of the line segment. The trace is blanked at this time so the dot does not appear on the screen. Next, X-STORE then

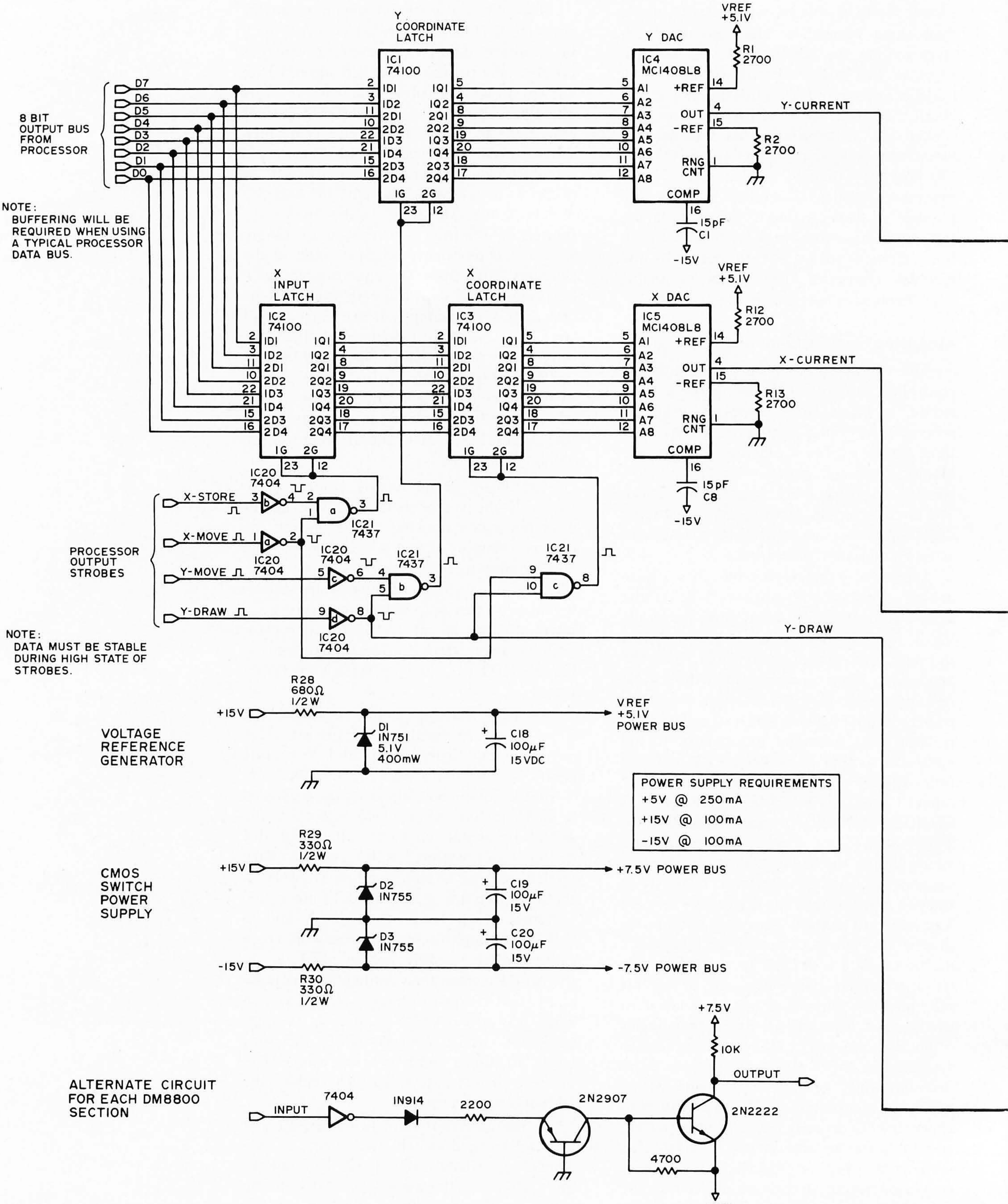
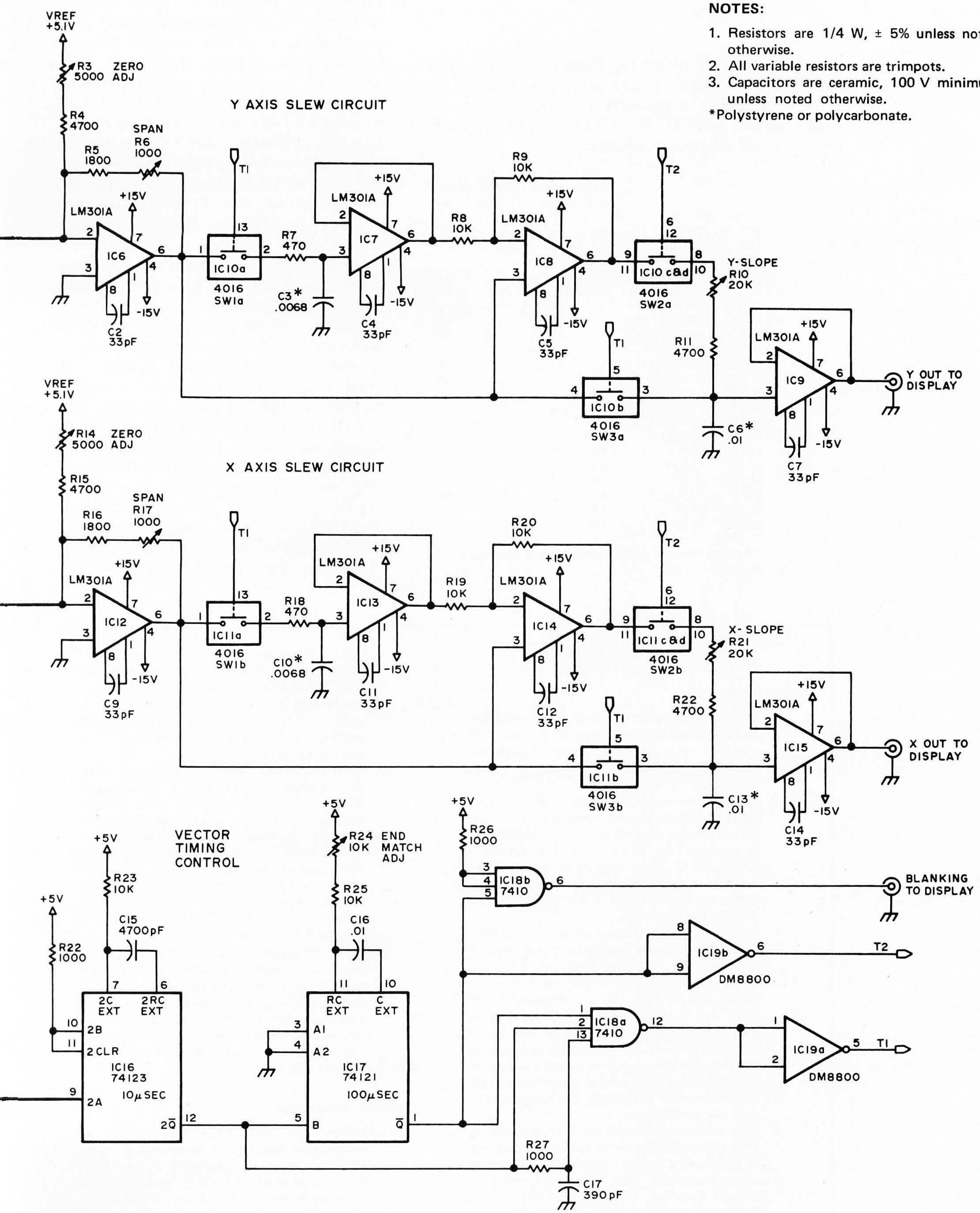


Figure 2: Schematic Diagram of a Vector Graphics Controller. This circuit will produce signals with a range of −2.5 V to +2.5 V for use in driving a standard XY oscilloscope with bandwidth greater than 100 kHz. The power supply wiring of digital

NOTES:

1. Resistors are 1/4 W, ± 5% unless noted otherwise.
2. All variable resistors are trimpots.
3. Capacitors are ceramic, 100 V minimum unless noted otherwise.

*Polystyrene or polycarbonate.



integrated circuits used in this circuit is summarized in table 1. (Analog integrated circuits have their power wiring indicated in the diagram. These circuits are listed in table 1 but have no power connections indicated.)

Table 1: Integrated Circuit List. This table gives each integrated circuit required in figure 2, along with its device type and, in the case of the CMOS and TTL digital circuits, power wiring pins.

		TTL power		CMOS power	
		+5 V	GND	−7.5 V	+7.5 V
IC1	74100	24	7	–	–
IC2	74100	24	7	–	–
IC3	74100	24	7	–	–
IC4*	MC1408-L8	13	2	–	–
IC5*	MC1408-L8	13	2	–	–
IC6*	LM301A	–	–	–	–
IC7*	LM301A	–	–	–	–
IC8*	LM301A	–	–	–	–
IC9*	LM301A	–	–	–	–
IC10	CD4016	–	–	7	14
IC11	CD4016	–	–	7	14
IC12*	LM301A	–	–	–	–
IC13*	LM301A	–	–	–	–
IC14*	LM301A	–	–	–	–
IC15*	LM301A	–	–	–	–
IC16	74123	16	8	–	–
IC17	74121	14	7	–	–
IC18	7410	14	7	–	–
IC19	DM8800	10	3	7	4
IC20	7404	14	7	–	–
IC21	7437	14	7	–	–

***Analog circuit connections are indicated in figure 2.**

Y-DRAW are executed. X-STORE loads the X coordinate of the endpoint into IC12 where it stays for the moment and does nothing. When a Y-DRAW is executed, the whole unit comes to life. Y-DRAW loads the Y endpoint coordinate into IC1 and transfers the contents of IC2, the X coordinate, into IC3, and both X and Y converters start to settle toward the endpoint value. Simultaneously with this action, switches SW1 and SW3 are turned off by T1, keeping the beam position where it was. After about 10 μs, switch SW2 is turned on by T2, the beam is unblanked and the output of the vector generator starts moving toward the endpoint values. This transition or drawing time is approximately 100 μs and is a function of the endpoint adjustment on IC17. It should be noted that the movement from the old value to the new value is an exponential function adjusted through the slope controls, but the lines between points will be straight since both X and Y are charging in a similar manner, though the velocity (hence brightness) of the beam does vary a bit from beginning to end. At the conclusion of the draw cycle, the beam is again blanked and the controller is ready to draw the next line. (The beam will be positioned wherever it

NOTES ON CALIBRATING THIS GRAPHICS CONTROLLER

Static Calibration

1. With all components except integrated circuits inserted and mounted, and presuming that you have no direct shorts, attach the controller to + and −15 V and +5 V and turn it on.
2. Using a meter, check to see that the right supply voltages are on the appropriate integrated circuit pins. Check to see that V_{Ref} is approximately 5 to 5.1 V and that the + and −7.5 V V_{DD} and V_{SS} supplies are correct. If substituting parts, don't use a zener any higher than 7.5 V since it could damage the CD4016s.
3. Insert only IC1 through IC3. Using either hard wire jumpers, switches or a computer program to drive the data bus lines B0 through B7, take note that when the bus inputs are all high, pins 5 through 12 of IC14 and IC15 are also high. Grounding all data bus inputs should result in low levels on these pins. It is a good idea to go down the bus one pin at a time to make very sure that all input lines end at the correct DAC pins, otherwise the digital to analog conversion will be incorrect.
4. Turn power off. Insert the DAC IC4 and op amp IC6. Attach a meter to read the voltage at pin 6 on IC6. Set the bus input switches or programmed data value to provide a binary code of 10 000 000 (200 octal). Turn power on and adjust the zero pot (R3) on IC6 until the output at pin 6 reads zero volts. Change the input code to 11 111 111 (377 octal) and adjust the span pot (R6) until the output is +2.50 V. Setting an input of 00 000 000 (0 octal) should result in −2.50 V out. The same procedure is used to calibrate IC5 and IC12. (Zero with R14, span adjust is R17.)
5. Turn off power. Insert IC16 through IC21. Temporarily, ground pins 1, 3 and 5 of IC20. Turn power on. Using either a pushbutton or cliplead intermittently ground IC20 pin 7 simulating a Y-DRAW strobe. A scope should be used to determine that IC16 pulses for about 10 μs and then triggers IC17 which lasts for about 100 μs. Control lines T1 and T2 should cycle between +7.5 V and −7.5 V corresponding to their TTL inputs.
6. Turn off power. Insert all the rest of the integrated circuits (be careful of the CMOS) and temporarily remove IC21. Reapply power and use the method in step 4, but take note that pins 6 on IC9 and IC15 are equivalent to IC6 and IC12 respectively. Turn off power, reinsert IC21. This concludes static checkout.

Dynamic Calibration

1. Unless you have super fast fingers on mechanical switches, a computer is necessary for dynamic checkout (while it is optional in static checkout). It is necessary to write a short program which outputs to the controller the full scale coordinates of a square with two diagonals (see figure 3).

was left from the last line segment, so it is possible to reduce software overhead considerably on continuous line graphs by just issuing a series of new endpoints.)

Maximum drawing rate can easily be determined for those who wish to use a faster processor. Each voltage converter requires about 3 to 5 μs to settle out and a draw cycle takes 110 μs. This means that 5 μs at least must be allowed between input of the beginning and endpoints and 110 μs must lapse before starting the next line segment. With an 8008 such as I used, there is little possibility of driving the controller too fast, but with an 8080A system a delay loop may be necessary between segments. The next step is to do some cartooning with software.

Obviously before we can get too carried away discussing graphics software you must get your controller calibrated and working. The procedure outlined in "Notes on Calibrating This Graphics Controller" should result in the best chance for success.

Now How Do I Use It?

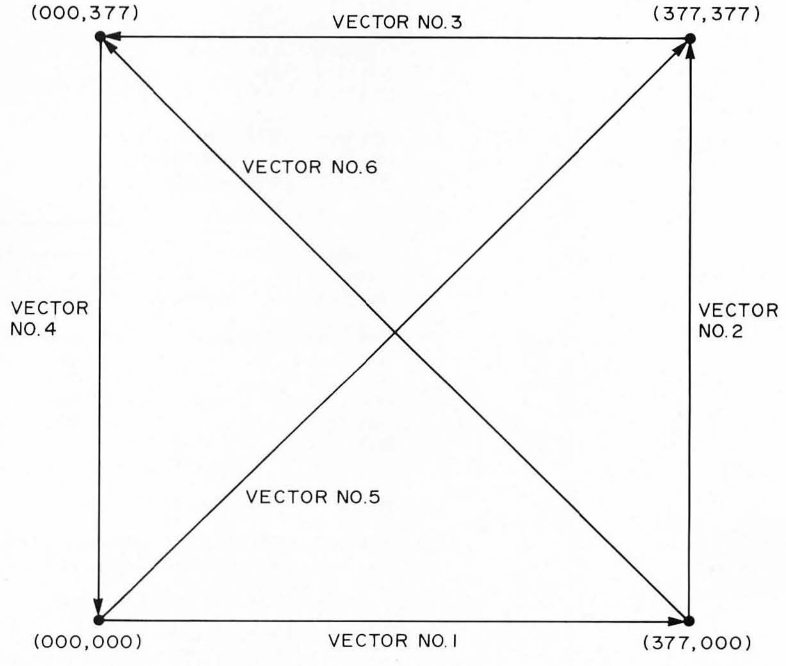
Now that the hardware of vector graphics has been conquered, you should be ready to write some simple display routines. Anyone owning a home computer system is always at a loss to justify it to friends and family who continually ask what purpose it serves. By using a little psychology and realizing that these people identify most often with Mom's apple pie, McDonald's and Star Trek, a little ingenuity can go a long way. Graphics goes a long way in impressing nonbelievers, especially when the computer is generating a picture of the USS Enterprise, shown in photo 1.

Drawing the Enterprise entails a program (see listing 2) which is very similar to that used for calibration (listing 1) except that the point list is longer so the program is written to accept variable length lists. There are many methods to program graphics. The easiest from the software point of view, but the worst for drawing speed and total number of lines, is to treat each line as an entity in itself. Four bytes would define the line segment as in listing 1's data table. This is required wherever it was placed on the screen, disregarding those displays which appear continuous as would be the case of a square or a triangle. As a specific example, 12 bytes are required to store the coordinates of a triangle when the three sides are treated as three line segments. Only 8 are required if a continuous drawing technique is incorporated. The problems arise when a display is made up of both disjointed and continuous lines. The program can be written to output one list of continuous

This square is all that is necessary to check out the remainder of the controller interface. (The X, Y and blanking outputs of the graphics controller have to be connected to an appropriate display. Any XY oscilloscope with a 100 kHz frequency response and an external horizontal input capability will suffice. The controller has to be connected to the output data bus and four decoded strobe lines.) The software which draws the calibration square is quite simple and consists entirely of outputting a list of coordinates stored in memory. The program of listing 1 was written for an 8008 system but is easily adapted to others.

2. If upon first turning on the display you obtain a perfect square with the diagonal meeting in the corners, go have a martini and relax, you're done (see figure 1c). It is more likely that the displays will only vaguely look like a square. If the line segments are too long or are too short and don't meet in the corners, adjust the end match adjustment, R24, on IC17, until perimeter segments (disregard the diagonals at this point) meet in the corners.
3. Next, fiddle with the slope adjustment pots (R10 and R21) until the diagonals meet in the corners. Any problems in accomplishing any of this is probably a result of the following: Frequency response of the op amps is too low (don't use anything slower than a 301A); the RC time constant of the slope adjustment is inadequate; your CD4016 crapped out; or your program is outputting incorrect coordinates.

Figure 3: Vector Graphics Test Pattern. The vector graphics generator is calibrated dynamically using a program like the one in listing 1, to produce a 6 segment square with diagonals at the extremes of the display grid of 256 by 256 points. In this figure, the vectors are identified, and octal coordinates of the endpoints are shown in parentheses.



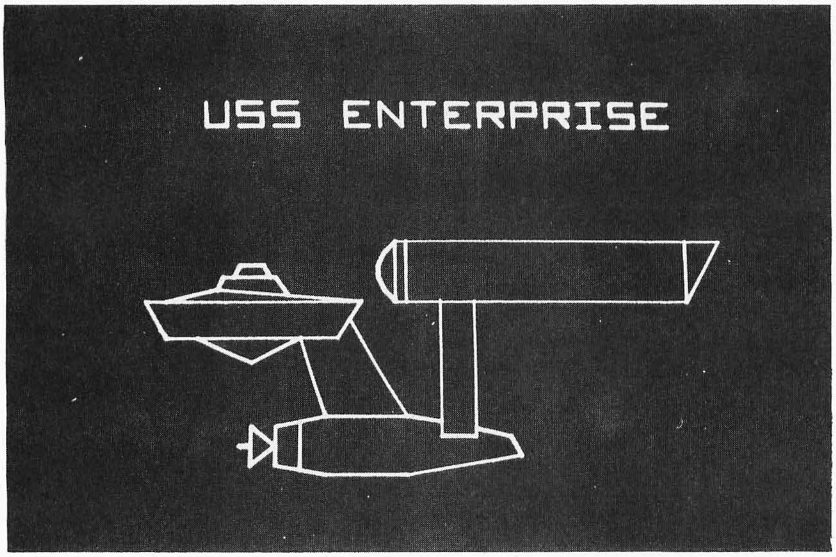


Photo 1: The author's Sanders 708, a surplus vector graphics display, shown portraying the Star Ship Enterprise using data of table 2 and figure 4. The lettering is not included in table 2.

lines and then jump to a list of separate line segments. This is the obvious method to accomplish graphics using a slow computer and lengthy display lists. Some problems arise when trying to translate such a picture from one point to another. A way of dealing with the problem of translation of picture elements from one location to another was suggested by Richard Lerseth in his article, "A Plot Is Incomplete Without Characters" on page 64 of July's BYTE. His technique is to use picture elements specified as a series of relative coordinates, to which the X and Y position coordinates are added at the time the image is displayed. (His method also involves economizing storage by using linked chains, but that is a slightly different issue.) When using a fast processor such as an 8080A or Z-80, the time required for a routine to do simple 8 bit additions for X and Y coordinates should be about the same (order of magnitude, 100 μs) as the vector drawing time of the circuit in figure 2. For those of us who have slower first generation processors, use of separate line segment lists with absolute coordinates calculated in advance may be a requirement if the screen is to show a complicated picture without

Listing 1: An 8008 Program to Generate the Calibration Pattern. The listing uses original Intel 8008 mnemonics; the data table at the end of the listing contains two bytes of starting coordinate followed by two bytes of ending coordinate for each point in the pattern.

Split Octal Addr	Octal Code	Label	Op	Operand	Commentary
001/000	056 001	START	LHI	H(TABLE)	Page address, data table;
001/002	066 040		LLI	L(TABLE)	Byte address, data table;
001/004	307	XFERLOOP	LAM		A := XMOVE value;
001/005	143		OUT	XMOVE	Perform XMOVE output;
001/006	060		INL		Point to YMOVE value;
001/007	307		LAM		A := YMOVE value;
001/010	147		OUT	YMOVE	Perform YMOVE output;
001/011	060		INL		Point to XTORE value;
001/012	307		LAM		A := XSTORE value;
001/013	145		OUT	XSTORE	Perform XSTORE output;
001/014	060		INL		Point to YDRAW value;
001/015	307		LAM		A := YDRAW value;
001/016	151		OUT	YDRAW	Perform YDRAW output;
001/017	060		INL		Point to next YMOVE value;
001/020	306		LAL		A := L [check end address low order];
001/021	074 070		CPI	L(TABLEND)	Is end of table reached?
001/023	110 004 001		JFZ	XFERLOOP	If not then continue loop;
001/026	104 000 001		JMP	START	Restart pattern when done;

Split Octal Address	Label	octal coordinates: start		octal coordinates: end		Commentary
		XMOVE	YMOVE	XSTORE	YDRAW	
001 040	TABLE	000	000	377	000	Vector #1;
001 044		377	000	377	377	Vector #2;
001 050		377	377	000	377	Vector #3;
001 054		000	377	000	000	Vector #4;
001 060		000	000	377	377	Vector #5;
001 064		377	000	000	377	Vector #6;
001 070	TABLEND	xxx				

Notes:

XMOVE, YMOVE, XSTORE and YDRAW are output port assignments of the author's system, corresponding to the strobe lines of figure 2.

All constants in this listing are octal.

H(TABLE) is high order 8 bit field of address of TABLE; L(TABLE) is low order 8 bit field of address of TABLE. The two high order bits of H(TABLE) are ignored.

flicker. However, a surprising number of lines can be displayed before flickering occurs when using the separate line technique.

To write a stationary display routine, it is necessary to determine the coordinate points of the picture's vectors. Graph paper which has been ruled for 377 by 377 octal squares (or hexadecimal if that's your preference) is not easily obtained, so be ready to count squares and label ordinary graph paper. Figure 4 illustrates the technique of creating a display table using graph paper. The illustration is drawn first, using a ruler or other straight edge to provide the lines. A pencil is a must at this stage, since erasure is a very real requirement when working out any sketch. Once the drawing has been finalized, octal coordinates are added to the basic graph paper pattern. In the case of the Enterprise, I wanted to fill the screen, so I scaled the coordinates to use most of the display area in the horizontal (X) direction, and used the same scaling in the vertical (Y) direction. Then, the vector list is created by going over the drawing line by line and determining the four coordinates required. The small numbers near the line segments in figure 4 identify the octal addresses of these line segments in the vector table, table 2. When the data for a picture is all encoded from the graph paper, it can be loaded into your computer's memory. If your processor is an 8008, then the program of listing 2 can be loaded and executed; otherwise, you'll have to create an equivalent program for your own processor. (The data of table 2 remains the same regardless of the processor used.)

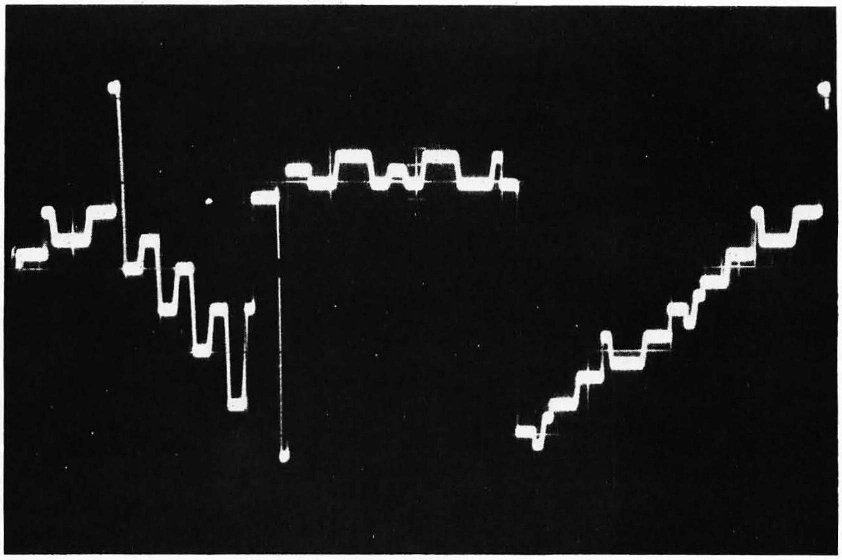


Photo 2: A scope trace of the X coordinate output of the circuit in figure 2 during the display of a typical picture. The trace is 3 V full scale and is approximately 15 ms long. The transition time from one voltage to the next is fixed by the hardware at 100 μs (approximately 1/150th of the horizontal width of this picture).

What's Next?

There is a fundamental problem with the home computer system or should I say the home computer experimenter. The systems keep getting bigger and more expensive.

Continued on page 120

Listing 2: An 8008 Program for Drawing Lengthy Pictures. This program is similar to the program in listing 1 with the addition of a generalized end of data test. In operation, it will be the sole program active in the 8008, continuously cycling through the vector table which starts at location 001/050.

Split Octal Addr	Octal Code	Label	Op	Operand	Commentary
001/000	056 001	START	LHI	H(TABLE)	Set up the memory pointer
001/002	066 050		LLI	L(TABLE)	to the beginning of table;
001/004	307		LAM		A := XMOVE value;
001/005	143	RFSHLOOP	OUT	XMOVE	Perform XMOVE output;
001/006	060		INL		Point to YMOVE value;
001/007	307		LAM		A := YMOVE value;
001/010	147		OUT	YMOVE	Perform YMOVE output;
001/011	060		INL		Point to XSTORE value;
001/012	307		LAM		A := XSTORE value;
001/013	145		OUT	XSTORE	Perform XSTORE output;
001/014	060		INL		Point to YDRAW value;
001/015	307		LAM		A := YDRAW value;
001/016	151		OUT	YDRAW	Perform YDRAW output;
001/017	060		INL		Point to next value;
001/020	110 024 001		JNZ	NOINCH	If no overflow then test end of table;
001/023	050		INH		Else increment high order address;
001/024	307	NOINCH	LAM		A := next value [may be XMOVE or END];
001/025	074 377		CPI	377	Is it END of data value?
001/027	110 005 001		JNZ	RFSHLOOP	If not then continue inner loop;
001/032	104 000 001		JMP	START	Else restart the program;

BYTE'S BUGS

Here lies documentation of known bugs detected in previous editions of BYTE . . .

Conversion Patch

Regarding the September 1976 issue, the following points were phoned in by Bob Van Valzah, 1140 Hickory Trail, Downer's Grove IL.

On page 58, in listing 4b of James Brown's article "How To Do A Number of Conversions," a typographical error occurred at address 0021. The correct code and comment is:

```
0021  6F        MOV   L,A
```

Bob also points out that zero suppression works "too well" with the decimal routine as designed by author Brown: It suppresses a zero value completely. Bob suggests the following patch replacing line 44 (and pushing TENSTABL down in relative address space):

```
MOV     A,C     get NONZERO flag;
ORA     A       test and set flags;
RZ              original return if some digits
                printed;
MVI     A,'0'   output of an ASCII zero
JMP     PUT       character and return if all
                  blank so far;
```

To use this patch, the reference to TENSTABL will have to be modified at location 0001.■

Patch of a Patch

The suggested biorhythm program patch given on page 100 of your July issue contains an error. This patch will, in fact, initiate all cycles in phase beginning at one's birthdate; but, if used as specified, errors occur in the incrementing of dates on the chart. By deleting lines 130 to 470, the loop which reads data into the vector T(I) is deleted. Data must be read into this vector in order for the chart dates to increment properly. I therefore suggest that the following patch be added to the previous patch.

```
122 FOR I = 1 TO 12
124 READ T(I)
126 NEXT (I)
```

This may save problems for some of your readers.

Gary Sleater
302 E El Caminito Dr
Phoenix AZ 85020 ■

The Price Is Wrong

According to IMS Associates, the price of the IMSAI floppy disk drive and interface controller mentioned on page 83-84 of BYTE's September 1976 issue is $1919 assembled, $1719 unassembled; additional drives without a cabinet are $1095. The prices quoted in "What's New?" are incorrect.■

Here Are Design Equations

In Bruce Filgate's article, "A Morse Code Station Data Handler," October 1976, we omitted the design equations for the NE567 tone decoder which should have been printed in figure 6, page 70. To remedy the situation, here they are:

Center Frequency: F ≅ 1 / (R1 * C1)
Band Width: BW ≅ 1070 * SQRT (VI / (F * C2))

where:

R1, C1 and C2 are as defined in figure 6, page 70 of October BYTE
VI is the input voltage to the circuit
F is the center frequency.■

Refining the Plot

Author Richard Lerseth reports that he received a letter from J C Rucklidge, associate professor of geology at the University of Toronto, Ontario CANADA. Prof Rucklidge used Richard's article ("A Plot is Incomplete Without Characters" in July 1976 BYTE) to program character outputs from his system. In so doing, he verified the accuracy of the printed tables, but found one minor error in equations 16 and 17 on page 70. These equations should read:

X = XO + H * DHX + (V − 5.0)* DVX (16)

Y = YO + H * DHY + (V − 5.0)* DVY (17)■

Caveat Programme

Past history has shown that occasional typesetting or logic bugs occur in programs printed in BYTE. Readers who find such bugs are urged to claim a bit of fame in this column by sending us the evidence, and suggested patches. Two examples have been documented here this month.■

A Bug Way Back

Boyd S Miner, K4KEP, editor of *Bits and Dits*, the newsletter of the Carolina Computer and Radio Amateur Association, points out an error in the statement of amateur radio Teletype mark and space frequencies in the September 1975 issue of BYTE, page 32:

Don Lancaster puts out some real good articles but in "Serial Interface" in BYTE September 1975, page 32, he states that Ham radioteletype audio frequency shift keying (AFSK) 2125 is mark and 2925 is the space frequency. This "tain't so." The mark and space tones are multiples of 425 Hz. In fact they are the 5th and 7th harmonics. Therefore the mark is 2125 as stated but the space is 2975, not 2925. At the time these standards came into being the FCC required 850 Hz shift. ■

A Note about Borrowing Routines from EDUCATOR

The EDUCATOR program from the Digital Group is an extremely effective way to learn what takes place within the processor when you are learning machine level programming.

The EDUCATOR-8080 article in the July 1976 BYTE *[page 22]* appears to contain at least two errors. I discovered these when attempting to use some of its text display routines.

The article neglects to mention that the user defined subroutine CHRPR (character print) has to add 200 (octal) to the code in the accumulator to get a valid ASCII character.

The subroutine CHEDT loops indefinitely on the first string of blanks. The HL register needs to be incremented after a string of blanks is output. I inserted an INX H (043 octal) between the present lines 000/310 and 000/313 appearing on page 26 of the July issue.

Since the listings published are obviously not exactly those used in the Digital Group software, the readers should be advised that further debugging may be necessary.

Dr Charles F Douds
281 Poplar St
Winnetka IL 60093 ■

Clubs and Newsletters

KIM-1 USER NOTES

At the Midwest Regional Computer Convention held in Cleveland, a group of KIM-1 users had the opportunity to meet with Rick Simpson of MOS. The group agreed that there was a need for KIM-1 owners everywhere to be able to exchange ideas much like other user groups, and that a newsletter would act as a vehicle for that purpose.

KIM-1 USER NOTES will be published every 5-8 weeks. Cost for subscriptions will be $5 for the next six issues (including first class postage). This rate is for US subscribers only. Make checks and money orders payable to: KIM-1 c/o Eric Rehnke. No cash please. Write: *KIM-1 USER NOTES*, c/o Eric Rehnke, Apt No. 207, 7656 Broadview Rd, Parma OH 44134.

South Florida Computer Group – I/O

In the August issue 1-4 it was announced that Dr Bruce Cameron will be the new editor of *I/O*. Bruce is a research biochemist with Papanicolaou Cancer Research Institute (PCRI or PAP). All inquiries, news items, articles, etc, should be sent to him at 1155 NW 14th St, Miami FL 33136; or phone (305) 324-5572.

The group has in the past sent complimentary issues to interested individuals, other clubs, and firms; but the club's mailing list has grown to such a point that now it's necessary to charge a fee of $2 per year for 12 issues. This will not affect the exchange of newsletters with other clubs.

Space Coast Microcomputer Club – Florida

This club at present (August) consists of about a dozen individuals who are interested in microprocessor-based computer systems. A few are employed at the John F Kennedy Space Center, while others are affiliated with Florida Institute of Technology and Florida Technological University. Many members also belong to the larger computer clubs such as the Southern California Computer Society and the Chesapeake Microcomputer Club. The purpose of this club is to further the usage of small computer systems through the mutual sharing of ideas and resources. Ray O Lockwood would be interested in hearing from you. His address is: 1825 Canal Ct, Merritt Island FL 32952, (305) 452-2159.

The New Orleans Hamfest and Computer Fest

On October 30-31, the New Orleans Hamfest and Computer Fest will be held at Archbishop Rummel High School, 1901 Severn Av, Metairie LA (a New Orleans suburb). This event is sponsored by the Jefferson Amateur Radio Club.

The Louisiana Council of Amateur Radio Clubs, the Louisiana ARRL Convention, MAR and other amateur radio groups are scheduled to meet. Banquets, noncommercial and other exhibits are being arranged. Call-in frequencies will be 146.34-.94 and 3.95 MHz.

Reservations and latest details may be obtained by writing to Dominick "Nick" Tusa, WA5RMC, chairman, New Orleans Hamfest, POB 10111, Jefferson LA 70181.

Byte Back – Ohio

Compute, Evaluate, Trade is the name of a new club in Ohio. Meetings are held in members' homes on the first Saturday of the month. There are no dues, other than talent and a desire to exchange information. Business applications are their special interest. Charles E Tyzzer Jr is the contact person, and he may be reached at (513) 268-6551 x587. The mailing address is: POB 104, Tipp City OH 45371.

Southern New England Computer Society

SNECS was formed at a meeting held in New Haven CT in July. Those present came from parts of Connecticut bordering on Rhode Island as well as Fairfield and the Hartford and New Haven areas. They represented a broad range of hardware, software, business, educational, and hobby interest in computers. A newsletter is in the works, and it will be called *YANKEE BITS*. Anyone wishing more information on this group should write: SNECS, 267 Willow St, New Haven CT 06511.

Ride the Data Bus – Michigan

Data Bus is the official newsletter of the Southeastern Michigan Computer Organization. Subscription is only available to persons living more than 50 miles from Detroit, at $6 per year (unless you're a member). *Data Bus* is also available on a newsletter exchange basis to other clubs. The editor is Jon Tara (313) 833-2847. The address is SMCO, *Data Bus*, 665 W Hancock No. 304, Detroit MI 48201.

Purdue Has PUNCH

Michael A Troutman, PUNCH president, informed us of the official existence of the Purdue University Computer Hobbyist Club. Mike says they have a constitution and are fully recognized as a student organization. The club meets weekly in the digital lab of the Electrical Engineering Building. The group is also planning lectures by key speakers for every other month. Their resources (those of EE) include two 6800 systems (Motorola and SWTPC), an 8080 system, a KIM-1 system, several CRT terminals, Teletype, high speed paper tape reader and access to the EE Dept PDP-11/70 timeshare system.

Membership is limited to students, staff and faculty; however, for a small activities fee, anyone is welcome to attend meetings. All interested persons may write to PUNCH, Room 67, Electrical Engineering Building, Purdue University, West Lafayette IN 47907.

Charlotte, North Carolina

A new club has been formed in the Charlotte NC area. Anyone interested in making contact with the Charlotte Microcomputer Society may call Dr Michael Allen

at (704) 597-2303 days; and Allen Grayson at (704) 366-7879 nights. Allen Grayson is interested in down to earth applications for computers. An example is the one he cited: A friend wrote a program creating a robot lawn mower. Now his friend sits back with his favorite beverage under a nice shade tree and watches his "robot" do the work.

Northern New Jersey

The Northern New Jersey Amateur Computer Group opens its membership to any interested person for an annual fee of $5. Meetings will be held from 6:30 to 11:00 PM the second Friday of each month at Becton Hall, Fairleigh Dickinson University, Rutherford Campus, Rutherford NJ. For further information, contact Murry P Dwight, Dwight Instruments Company, 593 New York Av, Lyndhurst NJ 07071. Phone: (201) 438-3334.

TRACE – Ontario, Canada

Toronto Region Association of Computer Enthusiasts has over 50 members in the greater Toronto, Hamilton, and Kitchener, Ontario areas. They meet once a month, usually on the first or second Fridays. For details concerning the club and newsletter, write: TRACE, Box 545, Streetsville, Ontario CANADA L5M 2C1.

Waterloo – Ontario, Canada

The Amateur Microprocessor Club of Kitchner-Waterloo, Ontario, has formed, with Charles Sooley acting as interim coordinator. The group has made some quantity purchases. For further information write: Amateur Microprocessor Club of Kitchener-Waterloo, Ontario, Canada, c/o Reading Room, Electrical Engineering, University of Waterloo, Waterloo, Ontario CANADA N2L 3G1.

Auckland, New Zealand

Info File of Mt Roskill has formed a group of N2 users and is asking help in getting off the ground. They need schematics, PC board layouts, and software. Newsletters would be appreciated. Write: Info File, POB 27-206-B, Mt Roskill, Auckland NEW ZEALAND.

Teen Hobbyist

Jeffrey Rice, 16, of 303 Wilson St, Elkins WV 26241, would like to join or form a computer club in his area, if there is any interest.

Enterprising Jeff, a would-be computer hacker, is working two paper routes and holding a part time job assembling boards for a local microcomputer distributor. He is saving up to buy a 8800 or 6800 based system while attempting to educate himself in computer science. He writes that between the BASIC he's learning on the PDP 8/E system, the minimal configuration COSMAC computer he is building, and as much literature as he can obtain, along with his "hands-on" hardware experience with his job, he's doing OK. You can contact him at the above address.

Ottawa – Ontario, Canada

We have received word that a group calling itself the Ottawa Computer Club has formed. At the present time the only information we have is that the meeting schedule is posted at Pettazzoli College Residence, located at Rideau and Chapel Av, Ottawa, monthly. They are looking for new members, and if you would like to play Sherlock Ohms, we suggest you investigate.

Curriculum Developer Needs Inputs

Robert F Tinker has founded Technical Education Research Center, located at 575 Technology Sq, Cambridge MA, phone (617) 547-3890. He is interested in creating curricula in the emerging technologies of low cost electronics for student use. A newsletter, *Hands On!*, is proposed as a quarterly transmission line for practical laboratory oriented educational ideas with a bias toward the uses of electronics and computers.

Denver Amateur Computer Society

DACS Newsletter, Volume 2/1, has mushroomed. The July issue was expanded to 44 pages. Some of the changes include longer features, new products, users' reports, tips, and "Dear Nina," letters from wearied and/or wearisome wives to the editor. The club is also holding a contest to come up with a name for the publication; the prize is a year's free membership. Entries, along with subscription orders ($9) should be sent to Jim Clark, *DACS Newsletter*, POB 6338, Denver CO 80206.

RPI ACM Chapter (Capital District NY)

Jerry Irving of RPI writes: The Rensselaer Polytechnic Institute student chapter of the ACM is forming a special interest group for people who are interested in minis and micros. We have had one meeting to see how much interest there would be and found a great deal of it. The group will be open to anyone, not just students. Those in the

capital district area of New York should be just a short drive from RPI. There are many minicomputers on the RPI campus and a number of people at the first meeting work with these, rebuilding, programming or writing software for them. There are people in the area who were at the first meeting who are starting to build their own systems. This group is forming in order to supply a place for people to meet to compare their systems, exchange ideas, and give them the advantages of being in a nationally recognized organization. Once under way we will have regular meetings and try to have speakers and demonstrations as often as possible.

We plan on having our organizational meeting to set up the foundation of the group in either August or September.

Anyone interested in coming or joining should write to the RPI-ACM, Amos Eaton Hall, RPI, Troy NY 12180, or call either Jerry Irving (518) 756-6827 or Will Strang (518) 274-8361 for information and directions.

The Computerist – NECS

The Computerist is an extension of the New England Computer Society. The club's newsletter is prepared on a text editor, and is printed by reduced photo offset. In

Volume 1/1 the goals are set forth: To serve all computerist and computer clubs in the New England region, report on club meetings, print a monthly calendar of regional events. It will also provide up to date reports on new products, publications, local suppliers, mail order dealers, and the like. It will also include a classified section, and both commercial and individual advertisements. *The Computerist* is an independently produced publication, which is the official newsletter of the New England Computer Society, and is supported in part by dues and subscription. Subscription rate is $6 per year. Robert Tripp is the contact man; his address is POB 3, South Chelmsford MA 01824.

Communication Co-operative for the Deaf and Hearing Impaired

The Communication Co-operative is a nonprofit organization establishing a time sharing computer system in the Northwest. Its purpose is to provide data processing services in order to help support a communications system for the deaf. This enables it to offer flexible computer services at reasonable rates and tailored to the needs of any business. Some of the services offered to businesses are inventory control, mailing lists, billing, and purchase orders. They also offer a 24 hour emergency service for the deaf, using Teletypewriter, and make available national and local news stored in a file and accessed by the Teletypewriter. If you would like to learn more, write: Communication Co-operative, POB 182, W Linn OR 97068.

Byte Out of Maple Address

"Someone took a byte (or two) out of the address for MAPLE, and I don't know whether all mail is arriving here," writes John Sikorski, president of "Microprocessor APL Enthusiasts," of the Clubs and Newsletters item on page 100 of September 1976 BYTE. The full address is:

MAPLE
Box 574 NUMS
303 E Chicago Av
Chicago IL 60611 ■

POLYPHONIC SYNTHESIZER BOARDS

ALTAIR 8800 & IMSAI BUS COMPATIBLE

TWO BOARD SET

Bd. 1 SRS-320
Generates any 4 notes in a 4 octave range that can be shifted up or down for a total range of 9 octaves!! (more boards-more notes)

Bd. 2 SRS-321
Modifies tones from SRS-320 and controls all parameters of the sound at the command of the computer.

PRICES
SRS-320 .... $175.00 }assembled
SRS-321 .... $175.00

send for free info pac

SRS Stillman Research Systems
P.O. Box 14036
Phoenix AZ 85063

Dealer Inquiries Welcomed
prices subject to change w/o notice

WANTED

Microprocessors Kit or Assembled

ALTAIR • SPHERE • MOTOROLA
INTEL & OTHERS
Altairs, immediate delivery.

Memories, I/O boards, chassis, power supplies, modems, peripherals, specials, prom programmers, ADCS, DACS, GPIOS, RS232 faces, and any associated software. Ask about our complete Altair system. Available now.

FORWARD INFORMATION AND/OR INQUIRIES TO:

AMERICAN USED MICROPROCESSORS, EQUIPMENT & SUPPLY CORP.
AT THE CHICAGOLAND AIRPORT
BOX 515
PRAIRIEVIEW, ILLINOIS 60069
TELEPHONE (312) 634-0076

For complete monthly catalog of items available and in stock Enclose $10. for six issues. Many new unused items. Amazing values for everyone. Tested and guaranteed.

Computer Mart of New York Inc.

NOW ON LONG ISLAND

IMSAI, SWTPCo, Digital Group
Processor Tech, Apple, OSI
TDL-Z-80, Seals, Cromemco,
Sphere, Tarbell, Oliver

Magazines, books, chips, sockets, connectors, terminals.

IT'S ALL HERE WAITING FOR YOU
FRIENDLY ADVICE TOO

New York City
314 5th Ave
(32nd St)
212–279-1048

Long Island
2072 Front St
East Meadow NY
516–794-0510

COMPUTALKER

MODEL **CT-1**

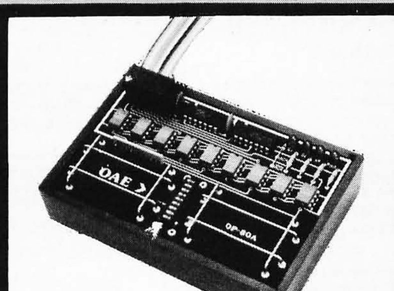
SPEECH SYNTHESIZER

High Quality Voice Output
Altair/IMSAI/Polymorphic plug-in
Nine Acoustic Control Parameters
Includes 8080 Interface Software
$395.00 plus shipping

Order a demonstration cassette
$2.50 postage paid

Write for informative literature

Calif. residents add 6% sales tax
COMPUTALKER CONSULTANTS
P.O. Box 1951, Santa Monica, CA 90406



Pictured above is the new OP-80A High Speed Paper Tape Reader from OAE. This unit has no moving parts, will read punched tape as fast as you can pull it through (0-5,000 c.p.s.), and costs **only $74.50 KIT, $95.00 ASSEMBLED & TESTED.** It includes a precision optical sensor array, high speed data buffers, and all required handshake logic to interface with any uP parallel I/O port.

To order, send check or money order (include $2.50 shipping/handling) to Oliver Audio Engineering, 7330 Laurel Canyon Blvd., No. Hollywood, CA 91605, or call our 24 hr. M/C-B/A order line: (213) 874-6463.

NOW!

A Computer Store in Burlington, MA

Offering the complete line of *Altair mainframes and peripherals as well as boards & boxes, tools & instruments, books & manuals, supplies & media.

The Computer Store Inc.
120 Cambridge St.
Burlington, MA 01803
617-272-8770

***Altair is a registered tradename of MITS, Inc.**

For more information circle A 76

So you've got a personal computer. Now what?

101 Basic Computer Games

David H. Ahl. An anthology of games and simulations—from Acey-Deucey to Yahtzee, all in the BASIC language. Contains a complete listing, sample run, plus a descriptive write-up of each game. Our most popular book! Large format, 248 pp. $7.50 [6C]

What to Do After You Hit Return

Another collection of games and simulations—all in BASIC—including number guessing games, word games, hide-and-seek games, pattern games, board games, business and social science simulations and science fiction games. Large format. 158 pp. $6.95 [8A]

Fun & Games with the Computer

Ted Sage. "This book is designed as a text for a one-semester course in computer programming using the BASIC language. The programs used as illustrations and exercises are games rather than mathematical algorithms, in order to make the book appealing and accessible to more students. The text is well written, with many excellent sample programs. Highly recommended." —*The Mathematics Teacher* 351 pp. $5.95 [8B]

Game Playing With the Computer, 2nd Ed.

Donald Spencer. Over 70 games, puzzles, and mathematical recreations for the computer. Over 25 games in BASIC and FORTRAN are included complete with descriptions, flowcharts, and output. Also includes a fascinating account of the history of game-playing machines, right up to today's computer war games. Lots of "how-to" information for applying mathematical concepts to writing your own games. 320 pp. 1976 $14.95 [8S]

BYTE Magazine

If you are considering a personal computing system now or later, BYTE provides a wealth of information on how to get started at an affordable price. Covers theory of computers, practical applications, and of course, lots of how-to build it. Monthly. 1-Year sub'n $12.00 [2A], 3-Years $30.00 [2B]

Games & Puzzles Magazine

The only magazine in the world devoted to games and puzzles of every kind — mathematical, problematical, crosswords, chess, gomoko, checkers, backgammon, wargames, card games, board games, reviews, competitions, and more. Monthly. 1-Year sub'n $12.00 [3A]

Games With The Pocket Calculator

Sivasailam Thiagarajan and Harold Stolovitch. A big step beyond tricks and puzzles with the hand calculator, the two dozen games of chance and strategy in this clever new book involve two or more players in conflict and competition. A single inexpensive four-banger is all you need to play. Large format. 50 pp. $2.00 [8H]

Games, Tricks and Puzzles For A Hand Calculator

Wally Judd. This book is a necessity for anyone who owns or intends to buy a hand calculator, from the most sophisticated (the HP65, for example) to the basic "four banger." 110 pp. $2.95 [8D]

Creative Computing Magazine

So you've got your own computer. Now what? *Creative Computing* is chock full of answers — new computer games with complete listings every issue, TV color graphics, simulations, educational programs, how to catalog your LPs on computer, etc. Also computer stories by Asimov, Pohl, and others; loads of challenging problems and puzzles; in-depth equipment reports on kits, terminals, and calculators; reviews of programming and hobbyist books; outrageous cartoons and much more. *Creative Computing* is *the* software and applications magazine of personal and educational computing. Bi-monthly.
1-year sub'n $8.00 [1A], 3- years $21.00 [1B], sample copy $1.50 [1C]

The Best of Creative Computing — Vol. 1

David Ahl, ed. Staggering diversity of articles and fiction (Isaac Asimov, etc.), computer games (18 new ones with complete listings), vivid graphics, 15 pages of "foolishness," and comprehensive reviews of over 100 books. The book consists of material which originally appeared in the first 6 issues of *Creative Computing* (1975), all of which are now out of print. 324 pp. $8.95 [6A]

Computer Lib/ Dream Machine

Ted Nelson. This book is devoted to the premise that everybody should understand computers. In a blithe manner the author covers interactive systems, terminals, computer languages, data structures, binary patterns, computer architecture, mini-computers, big computers, microprocessors, simulation, military uses of computers, computer companies, and much, much more. Whole earth catalog style and size. A doozy! 127 pp. $7.00 [8P]

Computer Power and Human Reason

Joseph Weizenbaum. In this major new book, a distinguished computer scientist sounds the warning against the dangerous tendency to view computers and humans as merely two different kinds of "thinking machines." Weizenbaum explains exactly how the computer works and how it is being wrongly substituted for human choices. 300 pp. $9.95 [8R]

Artist and Computer

Ruth Leavitt, ed. Presents personal statements of 35 internationally-known computer artists coupled with over 160 plates in full color and black & white. Covers video art, optical phenomena, mathematical structures, sculpture, weaving, and more. 132 pp. $4.95 [6D] Cloth cover $10.95 [6E]

Computer Science: A First Course (2nd Ed.)

Forsythe, Keenan, Organick, and Stenberg. A new, improved edition of this comprehensive survey of the basic components of computer science. There has been an updating of important areas such as Programming, Structured Programming, Problem Solving, and other Computer Science Concepts. The quantity of exercises and problems has been increased. 876 pp. $16.95 [7D]

Mr. Spock Poster

Dramatic, large (17" x 23") computer image of Mr. Spock on heavy poster stock. Uses two levels of overprinting. Comes in strong mailing tube. $1.50 [5B]

Problems For Computer Solution

Gruenberger & Jaffray. A collection of 92 problems in engineering, business, social science and mathematics. The problems are presented in depth and cover a wide range of difficulty. Oriented to Fortran but good for any language. A classic. 401 pp. $8.95 [7A]

A Guided Tour of Computer Programming In Basic

Tom Dwyer and Michael Kaufman. "This is a fine book, mainly for young people, but of value for everyone, full of detail, many examples (including programs for hotel and airline reservations systems, and payroll), with much thought having been given to the use of graphics in teaching. This is the best of the introductory texts on BASIC."—*Creative Computing* Large format. 156 pp. $4.40 [8L]

BASIC Programming 2nd Ed

Kemeny and Kurtz. "A simple gradual introduction to computer programming and time-sharing systems. The best text on BASIC on almost all counts. Rating: A+"—*Creative Computing.* 150 pp. $8.50 [7E]

Problem Solving With The Computer

Ted Sage. This text is designed to be used in a one-semester course in computer programming. It teaches BASIC in the context of the traditional high school mathematics curriculum. There are 40 carefully graded problems dealing with many of the more familiar topics of algebra and geometry. Probably the most widely adopted computer text. 244 pp. $5.95 [8J]

A Simplified Guide to Fortran Programming

Daniel McCracken. A thorough first text in Fortran. Covers all basic statements and quickly gets into case studies ranging from simple (printing columns) to challenging (craps games simulation). 278 pp. $8.75 [7F]

Understanding Solid State Electronics

An excellent tutorial introduction to transistor and diode circuitry. Used at the TI Learning Center, this book was written for the person who needs to understand electronics but can't devote years to the study. 242 pp. $2.95 [9A]

Microprocessors

A collection of articles from *Electronics* magazine. The book is in three parts: device technology; designing with microprocessors; and applications. 160 pp. 1975 $13.50 [9J]

Microprocessors: Technology, Architecture and Applications

Daniel R. McGlynn. This introduction to the microprocessor defines and describes the related computer structures and electronic semi-conductor processes. Treats both hardware and software, giving an overview of commercially available microprocessors, and helps the user to determine the best one for him/her. 240 pp. $12.00 [7C]

Creative Computing Catalogue

Zany 12-page tabloid newspaper/ catalog lists books, magazines, art prints, and T-Shirts. A conversation piece even if you don't order anything. Free. [5A]

CREATIVE COMPUTING, Dept. B
P.O. Box 789-M, Morristown, NJ 07960

Please send me the following:

Quan.	*Cat.*	*Description*	*Price*

Books Shipping charge $1.00 USA, $2.00 Foreign ______

NJ Residents add 5% sales tax ______

☐ Cash, check, M.O. enclosed TOTAL ______

☐ BankAmericard } Card No. ______

☐ Master Charge } Expiration date ______

Name ______

Address ______

City ______ State ______ Zip ______

What's New?



A Suggestion is Fulfilled

In last month's new products section, a touch tone frequency generator was suggested as a possible portable product to be used as a portable data terminal. Well, a company called Executive Devices has produced a product which incorporates such a frequency generator and can be easily used in the portable terminal mode. This battery powered hand held device generates the full 16 combinations of tones possible with the touch tone encoding technique, and interfaces with the telephone network through an acoustic coupler. There are two models available:

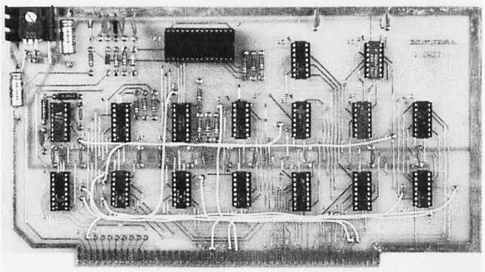
Pocket Data Terminal PDT-700 is $49.95, postpaid, and is the simple tone generator version. (Kit version $39.95.)

Pocket Data Terminal PDT-1000 is $89.95 postpaid, and is an extended

feature model with a nonvolatile user programmed memory of seven digits for automatic dialing of a frequently used number. (Kit version $69.95.)

Of course the auto dial product by itself won't do much for people stuck out in a country telephone exchange with relays and stepper switches for dialing, but any computer equipped with an auto answer function to a telephone port can receive and decode the touch tone frequencies of this device. For the home hacker, this can be accomplished by tapping into the audio section speaker outputs of a commercial ATT approved phone answering device. The auto answer logic is built into such a phone answering device, so all the computer has to do is to test the cassette motor control line and listen through a bank of tone sensitive filters for the frequencies involved. According to verbal reports, several individuals are already at work automating such home functions as lighting control through remote ports accessed in this manner; there ought to be a market niche for the commercial answering equipment needed (to plug into an Altair or other computer main frame).■

Real Time Programming, Anyone? (or How to Make a Perpetual Audio Visual Date Book)



COMPTEK, a company located at POB 516, La Canada CA 91011, has come up with just the thing needed to implement the application in the subtitle of this new products story. In order to implement a real time process such as an appointment book, some means of noting time and date digitally for each hour of the day, every day of the year, is required.

The way to accomplish this time measurement is to use this $98 real time clock board for the Altair or IMSAI computers. The board contains a conventional LSI clock chip, the National Semiconductor MM5318N shown in the photograph (the largest plastic package integrated circuit). The clock interfaces to the Altair in either a programmed IO mode, or an interrupt IO mode of operation. For clock interrupts, the board can be programmed to interrupt 60 times per second, once per second, once every 10 seconds, once per minute, once every 10 minutes (radio amateurs take note), once per hour, or twice per day. Also programmable through the board are the fast setting modes of the clock chip.

The microcomputer system treats the clock as a peripheral, and reads the time directly in a 24 hour HH:MM:SS format, BCD encoded. For the user wanting to implement the date book, some software information management structures will of course have to be created to use it, but the hardware of this board is a key element which makes such a system possible. Imagine: your computer with its audio and video output devices nagging you to get up, go on vacation, keep the appointment with the dentist, etc.■

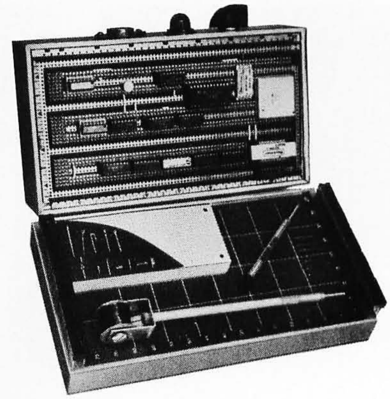


Is This a Programmable Calculator or Computer System?

Here is the latest Hewlett-Packard innovation, the flexible disk drive for the desk top calculator. The disk drive plugs into the HP 9825 desk top programmable calculator, to provide mass storage capability far beyond the built in data cartridge drive. The new product comes in two forms. The 9885M master disk drive is priced at $3900, and the 9885S slave (up to three of which may be used with one 9885M controller) is priced at $2500. The 9825 can handle eight of the 9885M units, making it possible for the extreme case of 32 disk drives run by one calculator. Since each disk holds 468,480 bytes of data, in the extreme, the calculator could have 14,991,360 bytes of on line random access storage – hardly a mere desk top calculator's worth of mass storage.

The disk option will prove a most useful feature for this calculator, which is programmed in a high level language command system. The disk option extends the software of the calculator to include a directory feature for up to 320 named files on disk. All operations are done in a write-then verify mode to ensure reliability.

Hewlett-Packard is located at 1501 Page Mill Rd, Palo Alto CA 94304.■



Breadboard Kit Including Slit-N-Wrap Tool

Vector Electronic Co, 12460 Gladstone Av, Sylmar CA 91342, has introduced this new modular breadboarding kit containing 36 different parts. The list includes terminals sockets, perforated circuit boards, wire, tools and many other items designed for custom wrapped wire circuit construction. The kit is designated 41X by the company, and the total cost of parts for the kit is lower than the catalog prices when ordered separately.

The kit contains the P180 Slit-N-Wrap tool to wrap and strip insulated wire found on a spool. This tool is a must for wrapping of extended bus structures, since there is no time consuming stripping or measuring to be performed. *[I'm using the P180 to complete a bus structure of backplane connectors on my homebrew system which has been partially assembled for months due to the inconvenience of the cutting and stripping with conventional wire wrap . . .* **CH***]*

In addition, the kit contains ten 16 pin DIP sockets, five types of wrap posts, and Vectorboard socket cards with and without printed circuit pads and buses. Also included are extruded aluminum frames to support the circuit board as a chassis.

The price of this omnibus kit of Vector's excellent products is $56.50, and it is available from stock. The 41X Multi-Conn Kit can be ordered directly from the factory (write to the attention of "Dept X"). Vector products are frequently found at retail computer stores, and retailers of the Vector line may find this an excellent "gift" product for the walk-in trade.■

E&L Instruments Increases Memory and Interface Capabilities of Their MMD-1 Microcomputer with an Accessory Board

A new accessory board model MMD-1/MI with extra RAM memory, Teletype interface, and audio cassette interface has been added to the E&L Instruments line. The MMD-1/MI simply plugs into the built in card edge connector on the MMD-1 and mounts on top of the unit as shown in the photo.

A MMD-1/MI accessory board completely assembled and tested, sells for $200. In kit form with all parts for assembly, the price is $150. The MMD-1/MI has the following features:

- 2 K byte programmable memory capability (1 K supplied).

ENIGMAS-1
COMPUTER GAMES IN BASIC
PRICE $8.00

ENIGMAS-1 IS A BOOK OF COMPUTER GAMES TAKEN FROM MY CATALOG
THE PROGRAMS IN THIS BOOK ARE:
GONE FISHING – GO FISHING TO MAKE MONEY
CONCENTRATION – MATCH THE HIDDEN NUMBERS–TWO CAN PLAY
SLOT-MACHINE – IT'S EASY TO LOSE YOUR MONEY
CRAPS DICE GAME
WITH DICE PRINT-OUT – SHOOT CRAPS AND SEE THE DICE
TANK ATTACK – TRY TO DEFEND YOURSELF
STAR SHIP
(STAR TREK TYPE GAME) – SHOOT SOME KLINGONS
SHERLOCK HOLMES
LOGIC GAME – CHASE PROFESSOR MORIARTY
ALL MY PROGRAMS IN BASIC HAVE BEEN WRITTEN TO RUN IN MOST ANY BASIC COMPILER-INTERPRETER WITHOUT ANY CHANGES.
THEY WILL RUN UNDER ALTAIR 4K OR 8K BASIC AND MOST ANY OTHER BASIC COMPILER-INTERPRETER PROVIDED YOU HAVE ENOUGH MEMORY TO RUN THEM.

PRICE $8.00 PPD. **SORRY NO COD'S**

B. ERICKSON
P.O. BOX 11099
CHICAGO, ILL. 60611

TURN ON WITH COMPTEK

Now your micro can control AC power outlets with **COMPTEK's NEW PC3200** Power Control System.

- 1 to 32 independently addressable control channels
- Remote power control units — no AC power on logic board
- Optically isolated, current limited, low voltage control lines
- Accessible through most BASIC's
- ALTAIR and IMSAI compatible

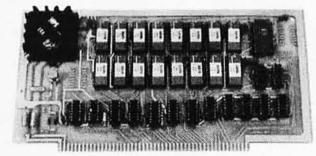
Write today for more details!

comptek

"Real World Electronics"

P.O. Box 516
La Canada, CA 91011

8,192 X 8 BIT STATIC MEMORY
EXCEPTIONALLY LOW POWER



KIT $295.00
*ALTAIR 8800/IMSAI 8080 BUS COMPATIBLE
*FAST 215 nS–FULL SPEED–FOR Z80 ALSO
*EXCEPTIONALLY LOW POWER–LESS HEAT
*LESS THAN OTHER "LOW POWER" MEMORY
*BATTERY STAND-BY CAPABILITY
*ALL SIGNALS TO MOS DEVICES BUFFERED
*SOCKETS FOR ALL IC'S

PROTOTYPING BOARD
LOW PROFILE IC SOCKETS
EDGE CONNECTORS
DB25 CONNECTORS

SHIPPING EXTRA, ADD $2.00
NJ RES. ADD 5% SALES TAX

ELECTRONIC CONTROL TECHNOLOGY
P.O. Box 6, Union, New Jersey 07083

JGM DEVELOPMENT LABS

P. O. Box 2345
W. Lafayette, IN 47906
(317) 463-7167

1/4 watt carbon film resistors.....$0.03
5 percent, 50 per value min.

1/2 watt carbon comp resistors...$0.02
10 percent, 50 per value min.

No-Nik wire strippers..............$16.00
10,12,14,16,18,21,23,25,28,31
34,37,44 or 54 thousanths
wire diameter

Carbide PC board drills

wire size:	55 - 65	$2.88
	66 - 70	$2.97
	71 - 75	$3.06
	76 - 78	$3.60
	79 - 80	$3.60

Spec sheets are available.

We would be pleased to send you a 'GOODIES' catalog listing all kinds of hard to find products including: Molex, Bishop Graphics, AMP, CDC manuals and forms, PC drills and mills, electronics tools, PC board manufacturing supplies, resistors, and semiconductors.

HEY, ALTAIR... GOT THE TIME?

If not, you need **COMPTEK's NEW CL2400** Real-time Clock.

- Self-contained hardware clock
- Can be set and read by BASIC
- Programmable interrupts
- Top quality board, components, and I.C. sockets
- ALTAIR and IMSAI compatible
- Uses: 24 hour clock
  Software timer
  Event timer

KIT — **$98** ASSEMBLED — **$135**

comptek

"Real World Electronics"

P.O. Box 516
La Canada, CA 91011

NOW!

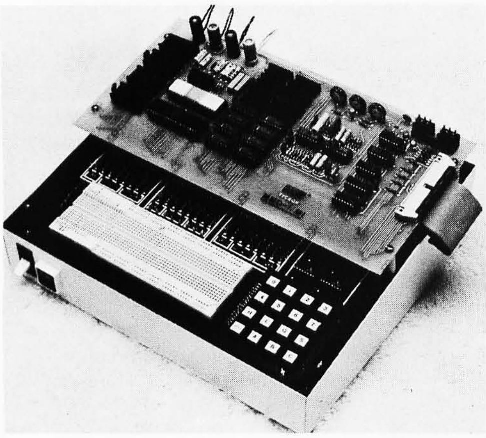
A Computer Store in Windsor Locks, CT

Offering the complete line of *Altair mainframes and peripherals as well as boards & boxes, tools & instruments, books & manuals, supplies & media.

The Computer Store
63 S. Main St.
Windsor Locks, CT 06096
203-871-1783

***Altair is a registered tradename of MITS, Inc.**

For more information circle A 103



- Teletype interface (20 mA current loop).
- Paper tape reader control for ASR33 Teletypewriters.
- Audio cassette interface.
- Sockets to accept up to 1 K PROM or ROM.

The addition of the MMD-1/MI board to the basic MMD-1 Microcomputer increases the system memory capacity to 2.5 K programmable memory and 1.5 K PROM or ROM. The Teletype interface and audio cassette interfaces allow easy and inexpensive data storage and retrieval. The MMD-1/MI is now available from E&L Instruments and from all E&L Instruments stocking representatives. E&L Instruments is located at 61 First St, Derby CT 06418.■

The Briefcase Computer Concept, Revisited

This neat little package of computing power, the DE68, made by the Digital Electronics Corporation, 415 Peterson St, Oakland CA 94601, is intended for use as a stand alone computer system in a briefcase. The system is completely self-contained, with the following characteristics:

- 6800 microprocessor.
- alphanumeric keyboard (96 character).
- miniature electronically controlled digital cassette system with 100 K byte on line capacity.
- 20 column wide alphanumeric display.
- 20 column alphanumeric printer (optional).



- 5.5 K byte PROM operating system with mnemonic translator, tape commands, memory manipulation features, breakpoints, single stepping of programs, traces, etc.
- 1 K bytes of user programmable memory standard, optional expansion to 4 K, 8 K or 32 K.

This is a completely self-contained microcomputer system which could be equally well used as a personal computer, an educational tool oriented to assembly level programming, or custom applications interfaced through its IO expansion connector (to the right of the processor card seen in the photograph). The DE68 price, completely ready to go as seen in the photograph, is $3500 (including the printer peripheral option) with delivery 60 days after receipt of order.■

Star Trek in BASIC?

Atlantis Consultants, POB 8493, Salt Lake City UT 84108, has come up with two Star Trek programs, written in BASIC, which are available now at $12.50 for the set. Included in the literature accompanying the programs, according to a letter from Earl Smith of Atlantis, are memory requirements, operational instructions, and tips and patches to help tailor the programs to any size system. These programs are accompanied by a one year warrantee against programming and typographical errors.■

Printed Circuit Board Hackers Take Note

Two new etched circuit board kits from Vector Electronic Co, 12460 Gladstone Av, Sylmar CA 91342, facilitate rapid production of quality circuit boards without expensive and time consuming processing with cameras and darkrooms. The kits, model numbers 32X-1 and 32XA-1, contain positive resist coated circuit boards, bare copper clad boards, and all materials necessary for fabricating circuit boards by the direct art then etch process, and also by the positive photo resist process.

For direct processing, circuit patterns are applied directly on bare copper laminated circuit boards, using the acid resistant rub transfer artwork sheets and other art materials contained in the kits. The sheets have patterns for 40, 24 and 16 pin dual in line packages, flat packs, lines in 5 widths, pads in 7 diameters, edge contacts in 2 sizes, alphabetic and numeric markings, and many miscellaneous coupling pad configurations.

The rub transfer patterns also may be used for making one to one size custom positives on the clear Mylar film supplied. This is then used to photo pattern the copper clad laminated boards coated with photosensitive etch resist. Vector's positive photo resist eliminates the need for photographic conversion of master artwork to negatives. The positive film is placed on the board, and exposed to a full or partial ultraviolet light source.

Exposure time requires approximately four minutes with either a GE "sun lamp," a commercial exposure lamp, or direct sunlight. A six ounce bottle of developer in solution accommodates about 600 square inches of sensitized surface.

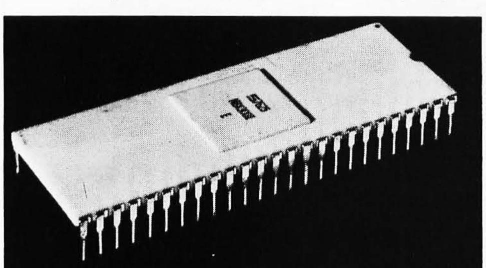
The exposed and developed boards are etched in the ferric chloride solution and chemical tray supplied. The simple developing and etching process takes about 20 minutes. Following washing, the board is ready for use.

Vector etched circuit kits are useful for both engineering and experimental circuit board fabrication. Existing full scale artwork may be copied directly with the supplied tracing and artwork materials.

The Vector 32XA-1 kit will make seven printed circuit boards, and is priced at $28. The 32X-1 sample kit makes two boards, and is priced at $11.50. ■

Need a Fast Microcontroller?

Signetics Corp, 811 E Arques Av, Sunnyvale CA, has introduced a new high speed microprocessor component called the 8X300, which is "optimized for control operations." This is a Schottky bipolar



processor with a simple two operand instruction set and a single instruction execution time of about 250 ns (operating on 8 bit data). The processor is available from stock at distributors for a quantity of 100 price of $86.75. No single quantity price was quoted, but based on comparable pricing strategies in the marketplace, we would expect the single unit price to a hacker to be below $250. What can a savvy BYTE reader do with a chip like this?

Well, look at the vector graphics problem, and consider design of an ROM programmed vector graphics processor. This machine is fast enough to do both the timing operations of a vector graphics generator, and the storage housekeeping for refresh. Build a custom machine for the 256 by 256 vector graphics application, with an intelligent refresh command language and a communications discipline for talking (as a peripheral) to your main microprocessor.

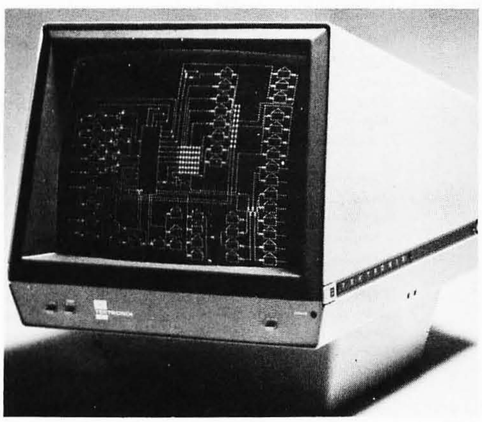
Take a look at the vertical microprogramming problem: emulating the instruction set of a known "nice" processor with lots of software. The 8 bit data paths of this machine, plus the high speed and internal scratchpad facilities should allow microprogramming hackers a very flexible alternative to the bit slice approach. According to the press release, it is "possible for 8 bit parallel data to be rotated or masked, to undergo arithmetic or logical operations, and then to be shifted and merged into any set of from one to eight contiguous bits at the destination – all within one 250 ns cycle." A glance

at the block diagram for the processor reproduced here corroborates this statement.

We'll be looking forward to some articles by readers on the use of this sort of chip to implement high performance homebrew rigs in the near future, to say nothing of the possibility that such a chip could be used as a key component of a true "high level language" machine implemented at low cost.■

Attention, Commercial and Industrial Graphics Users

Here is a state of the art graphics product designed for OEM customers only, the model 619 Storage Display Monitor made by Tektronix Inc, Beaverton OR. OEM only means that Tektronix is not interested in selling to "unsophisticated" users in this case. But as an example of a high resolution vector graphics display, all us individual users will be waiting with mouths watering for the introduction of smaller scale but high resolution units for personal systems. Here is what the state of the art price of $7125 (single quantity OEM) gets a systems engineer with graphics in mind these days:



The model 619 is a 19 inch (48 cm) direct view storage tube display which is designed for high density flicker free graphics. The 619 provides stored displays of combined alphanumeric and graphic information from analog sources and digital computers, with engineering data provided so that the OEM systems builder has full control in developing analog or computer interfaces to utilize the capability of the 619.

The storage tube of this device is used to achieve high density graphics without the need for providing expensive memory devices. But the bistable storage tube has two modes of operation: The storage mode allows conventional storage graphics which gives high density flicker free pictures built up over considerable periods of time; a second refresh graphics mode allows a partially stored and partially refreshed display to be generated. Only the fixed portion of the display is stored with storage graphics technology, and a variable refresh graphics portion covers the moving stuff. In a Space War game, this terminal would excel since an intricate background map could be constructed using the storage graphics technology of the instrument, with the variable graphics of the space ships, torpedos, players, etc, being provided by the refresh portion. The capacities of the machine in the refresh mode are 3200 inches of stored vectors and 630 inches of refreshed vectors. One can expect that those lucky BYTE readers who have access to a Tektronix 619 through a research, commercial or industrial job situation will have a field day with this machine for interactive games.■

Classified Ads for Individuals and Clubs

FOR SALE: Complete VIATRON System 21, includes printing robot, comm adaptor, also many tapes. Unit in perfect operating condition. Will consider throwing in the IBM Selectric Model 72 typewriter. Want $500 for all except Selectric. Will trade for video display board and memory for IMSAI, or small video terminal. Tom French, c/o Philips Data Systems, 3320 Holcomb Bridge Rd, Norcross GA 30092.

FOR SALE: Hewlett-Packard HP-185B oscilloscope with 187C dual trace plug in. Triggered sweep, DC-1000 MHz sampling. Internal DC level, 50 MHz sine wave, pulse calibration signals. One knob cracked but scope works good. $300. Mike Turner, Rt 2, Roberts St, Fletcher NC 28732. (704) 684-2112.

WANTED: I am looking for information on a FREIDEN "5610 Comptyper." (1) Pin outs on the aux output plugs of this unit. (2) Necessary data to interface this unit with a 8080 system. K9BJN P F Collins, 1611 W Madison St, Kokomo IN 46901. (317) 452-0265.

WANTED: New or used LA36 DECwriter II. I have written several programs in Altair BASIC. For information send $0.25 to Timothy G Roscoe, 153 Dorman Dr, Elizabeth PA 15037.

WANTED: Need schematics and manuals for DURA MACH 10, also tech data for ITEL System 3016SQD and/or DURA typewriter models 8123124 and 133038. I have considerable data (130 pages) on other models to swap or sell. I wish to communicate with other users and parts sources. Ronald Jenkins, 4490 Sirius Av, Apt 4, Lompoc CA 93436.

FOR SALE: IBM 2315 disk cartridge (S/N 22721) perfect condition; make offer. Rudy Wrobel, 12725 W 55 Terrace, Shawnee KS 66216.

FOR SALE: MITS 88-4MCD 4 K dynamic memory card. Fully assembled and tested by MITS $190; 88PIO parallel IO port with cable and connectors, assembled and tested $60. Complete set of Mark-8 boards with one extra memory board all brand new plus highly regulated Precision Systems power supply +5 V/8.5 A −12 V/2.0 A adjustable to −9 V, includes Radio Electronics Mark-8 blurb all for $110. Steve Caldara, 965 E State St, Ithaca NY 14850. (607) 273-9234.

FOR SALE: MITS Comter 256 terminal with acoustic telephone coupler, assembled and operational. Also, three 4 K dynamic memory boards and an 88-ACR cassette interface, all assembled. Steve Fritts, 2920 Bellevue St, Knoxville TN 37917, (617) 525-5407.

FOR SALE: I have a supply of new INTEL and TI 8080A microprocessor chips for $25 each. These can be used to update your Altair computer, to control intelligent peripherals, or for experimental circuits. Sockets and specs are available if requested. G Lorenc, Box 710 LU, Lockport IL 60441, (815) 838-4930.

WANTED: Low cost new or used terminal and/or acoustic coupler for private use. Write Rob Lufkin, Lodge 9, RFD 1, Box 52, Charlottesville VA 22901.

FOR SALE: Zilog Z-80 third generation microprocessor chips. $95 each including shipping and handling. We are a club and will purchase these chips from the manufacturer at quantity discount prices. We need a minimum of 25 orders. For a description of the Z-80 see the August 1976 *BYTE*, page 34. Send orders to Kristopher Kafka, 166 Marshland Rd, Apalachin NY 13732. (607) 625-2972. Sorry, no CODs, no credit cards.

FOR SALE: SCM computer terminal in perfect working order. Complete with cables and well-filtered power supply. All in working order. SCM No. 603601 schematic No. 4391171. Can easily be converted to IO hard copy printer. You pay shipping and $150. Will trade. Call or write Wallace Hurley, 2753 Parkside Dr, Fremont CA 94536. (415) 792-8589.

FOR SALE: Limited supply of core memories from major computer manufacturer with spec sheets. 8 X 64 X 64, $15; 16 X 64 X 64, $30; 17 X 64 X 64, $35; spec sheets, $0.75. Include shipping, excess refunded. Jim Kuiper, 542 Linden Ln, Ann Arbor MI 48103.

FOR SALE: FRIDEN printing terminal 1970, several available. 12.2 or 14.7 cps – full ASCII font – half duplex with breakin – 14½ paper; 140 print positions. Quiet, mechanically simple, cast aluminum housing. Size 21" X 23" X 11". Room for 15 3" X 5" PC cards. Parallel data input and output. Schematic for electronics to perform shift decode, repeating and logic interface with UART included (standard TTL). Mechanically overhauled, excellent operating condition $250 delivered, New York-Boston. J Colter, POB 122, Short Beach CT 06405.

WANTED: ITT Asciscope repair manual and schematics. George Ahmuty, 60111 Wendy Ln, Westport CT 06880. (203) 227-8534.

FOR SALE: Altair 8080, PTC Motherboard, IMSAI cassette interface, 4 K and 1 K RAM, 4 K EROM presently burned with assembler-monitor-editor, MITS quad parallel interface board, TI ASCII keyboard, PTC VDM-1 with expanded software and video monitor. Complete system: $1600. C G Hullquist, (615) 727-7793/8437, Box H, Mountain City TN 37683.

ENTREPRENEUR seeking creative electronics "wizard" as partner in new moonlighting business venture to develop computer products for high potential home and small business markets. Must be located in San Francisco area. If interested, please write R Lum, 20661 Marion Rd, Saratoga CA 95070.

FOR SALE OR TRADE: Friden Flexowriter Model SFD. Has 8-level printer, paper tape reader, paper tape punch and keyboard. Excellent condition. $300 or will trade for TVTI or TVTII. Call or write Bill Dawson, 316 21st St, Apt No. 4, Virginia Beach VA 23451. (804) 425-4411 (days), (804) 422-5921 evenings.

SWAP OR SELL: One (1) tape punch machine with keyboard, 75 cps punch and Burrough display. $400. Four (4) 75 cps punch mechanisms and driver cards. $75 each. Three (3) high performance stepper motor systems (2 motors and power supply and control logic). $300 each. Honeywell 3C cards – three (3) each: FC320 and BC320; one (1) DK320. Call after 5 PM (617) 369-2075. Albert Spitzak, 113 Everett St, Concord MA 01742.

FOR SALE: I have an HP-65 calculator to sell before I buy a computer, since having both would take too large a byte out of my savings. $700 buys it. Joseph R Danek, 3945 E Birchwood Av, Cudahy WI 53110. Call (414) 769-7489 between 9 and 12 AM CST.

FOR SALE: Issues 1-12 of BYTE, good condition. Best offer above cost of magazine. Robert Campbell, 4324 Walker St NE, Cedar Rapids IA 52402. (319) 393-4191.

FOR SALE: Altair 8800 computer, fully assembled and tested, with 1 K static RAM, $500 plus shipping and handling. R A Stook, Cherry Dr, Souderton PA 18964. Telephone: (215) 723-4766.

FOR SALE: MITS VLCT (Very Low Cost Terminal), complete except for case, $75; 4 K memory board for Altair 8800, fully assembled and working, $95; tape cartridges for IBM MTST typewriter, $5 each or swap for ? Used Altair 4-slot motherboard in reasonable condition, $10. Jim Einolf, 3900 Fox Dr, Loveland CO 80537.

FOR SALE: STELMA Model 2400 data modem. 1200 and 2400 bps over HF radio (used on APOLLO project for long haul data transmission). 2 units, spares, manual $250. Frank Gentges, 3512 Orme Dr, Temple Hills MD 20031. (301) 894-2613.

FOR SALE: MOS Technology KIM-1 microcomputer system. Purchased in March from MOS for $245; completely checked out. $175, including 5 V and 12 V power supply. Built to MOS specifications, and complete documentation. KIM-1 has hex keyboard, LED display, 1 K RAM, PROM-resident executive, TTY, audio tape, and BOS interfaces. Larry Hill, (617) 475-4700 or (617) 646-6428; 163 Waverly St, Arlington MA 02174.

FOR SALE OR SWAP: EIS telewriter-keyboard/printer unit, 8-level code (ASCII), 110 baud, full duplex, TTL level output. Operating condition, may need some adjustments. W/some data $295. Also some fully ASCII encoded keyboards $50. Money order to: Bill Dumbleton, 1713 Webster St, Omaha NE 68102. (402) 341-1529 days, 551-0577 evenings.

FOR SALE: 16 K words by 9 bit core memory modules (10" X 12" X 4"). Needs only T^2L interface and power supplies (+5, −5, and +30 volts). 1 μs read/write cycle time capability. Complete documentation available. For additional information write to Dave Bowman, 5437 W Tierra Buena Ln, Glendale AZ 85306, (602) 978-0332.

TRADE: Will trade by BYTE collection (issues 1-12, excellent conditon) for a complete set of QST magazines from January '75-June '76 inclusive. Jeff Stern, 809 Flanders Dr, Valley Strm NY 11581. (516),791-7471.

FOR SALE: FABRITEK core stack, model 999-6155-00-B 16 K words by 24 bits, no other data. Will include 34 type SN7528 sense amps. Best Offer over $200 takes it. Contact: Steve Marum, 116 Western Hills Dr, Apt 136-J, Howe, TX 75059. (312) 532-5292.

PARTNER(S) for microcomputer store wanted in Texas, Okla, Colo, or NM. I will finance up to one half interest in store; you must have direct microcomputer experience. Dr M E Reed 1417 Northgate Sq B-12, Reston VA 22090. (703) 437-5992.

FOR SALE: Assembled SWTP 6800 CPU board with support information consisting of *SWTP 6800 User's Manual, Motorola 6800 Programming Manual,* and *6800 Applications Handbook*. Will sell for $260 or best offer. Mark-8 CPU, Address latch, and Input multiplex boards, includes most TTL and resistors and caps, $30 or best offer. 18 slot card cage used – great condition, with 18, 22 pin WW and 17, 100 pin WW Amphenol card edge connectors, $25. Tim Tibbott, 714 N Nottawa, Sturgis MI 49091.

FOR SALE: Populated MIL MOD 8 boards: 8-2 TTY board $10. 8-3 Control board $10. 8-6, 8-7 IO boards $10 ea. 8-1 CPU board $18. Unpopulated 8-4 1702 type PROM board $5. C-MOD 8 MOD 8-9 board with 8316 2 K X 8 bit ROM containing 8008 operating system $30. Plus much information on the MOD 8 system. All boards built with PRIME parts. R DeHoff, Smith Hall, Kent OH 44242.

FOR SALE: High speed paper tape punch, Teletype Corp (BRPE) with maintenance manuals and parts manual. Eight channels at 110 characters per second. Very good working condition. Best offer. Ronald Howland, 23 Sumner St, Milford MA 01757.

AVAILABLE: Copies 1-9 of BYTE (two copies of issue 2). Any offer will be considered but would prefer to sell all in one transaction. Contact: John R Kaser, 7500 NW Peavy Arboretum Rd, Corvallis OR 97330. (503) 745-5703.

FOR SALE: 8008 based operating system featuring 11 human-engineered commands, allows full control of loading/examining memory, octal or hex dump, execution initiation, and read/write/control of cassette tape storage. Included are 47 pages of source code, operating instructions, and schematics for 256 level hardware pushdown stack, serial-parallel interface, PROM interface. $7.50 postpaid. If 15 orders are not received within 30 days from publication of this ad, your check will be returned uncashed. William E Severance Jr, Center Lovell ME 04016. (207) 925-2271.

Build This Video Display Terminal

Alfred I Anderson
721 15th Av SW
Rochester MN 55901

I first became interested in microprocessors about a year ago. After much thought, I decided to investigate this rapidly growing field. I did not have any particular goal in mind but I did want to learn how they work and what they can do. Therefore, I decided not to buy a kit but to build my own from scratch. I happened to select the Motorola processor for a number of reasons which I won't consider here. After a few weeks, I had a small system up and running. I even managed to interface an old "Tele-Term" that I borrowed from the Mayo Clinic where I work. Now (I thought) I had the cat by the tail (or the byte by the bit?).

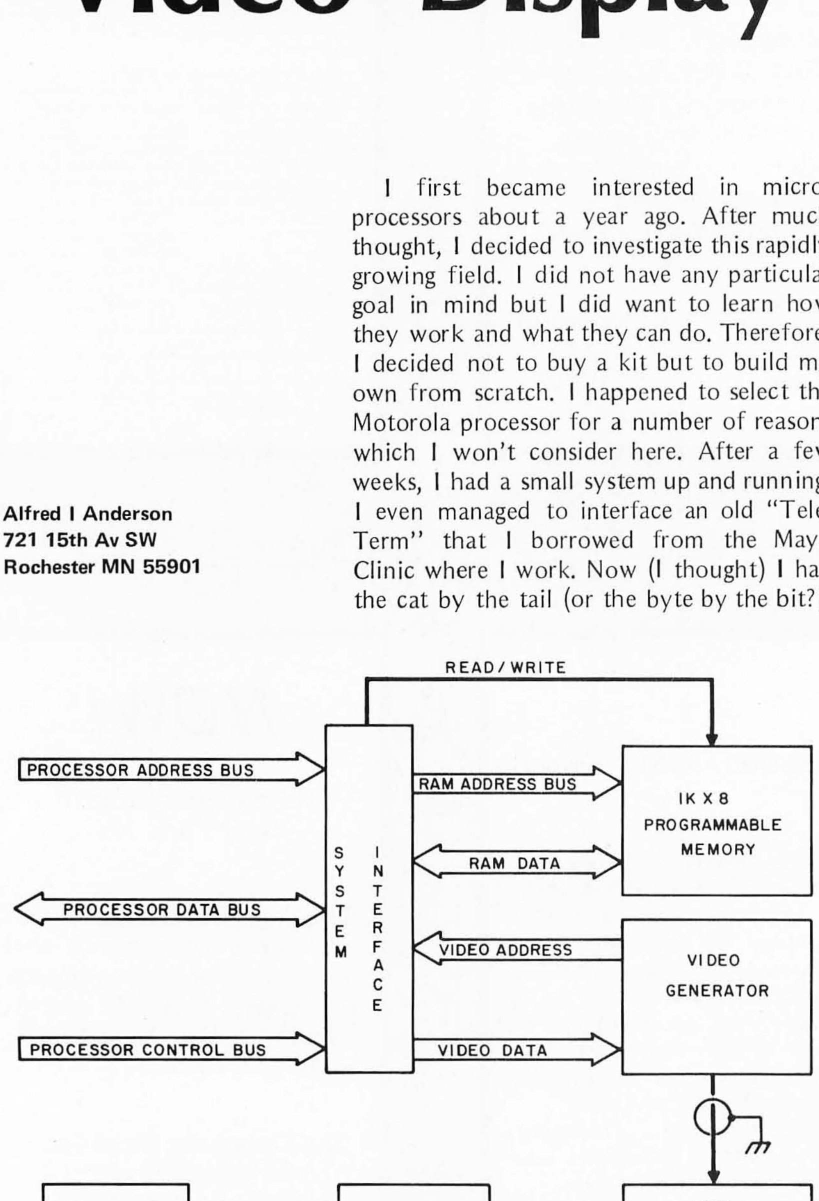


Figure 1: Video Terminal Systems Configuration. This block diagram shows how the terminal subsystem is organized as a display generator for the computer system.

It wasn't long until I became impatient with the slowness of the Tele-Term, its loud clacking, and the reams of paper I was going through. What did I need? A video terminal, of course. There are a number of good kits on the market place. My impression was that they are either too expensive for my homebrew budget or they are Altair compatible. Now I needed Altair compatibility for my homebrew system like a write only memory. It would not have been a trivial task to interface any of the affordable video modules designed for the Altair to my homebuilt 6800 system. What could I do?

The answer was in the June 1976 issue of BYTE. Mr Gantt's article "Build a Television Display" *[page 16]* provided the basic information I needed on television interfacing. I changed the circuit around a bit, added a few goodies, incorporated some memory, and wrote a few lines of software. I am now using a video terminal that you would like. The following paragraphs should outline everything you need to know to complete one of your own.

Design Specifications

For my purposes a video terminal had to have a number of features I deemed to be desirable. First, I wanted immediate access to any character on the screen without having to mess around moving little white boxes. This feature allows me, for example, to keep a real time clock display in the upper right hand corner of the screen. This video terminal uses a simple minded direct memory access technique enabling a complete screen update in the twinkling of an eye. I chose a screen configuration of 16 lines by 32 characters per line. This limited amount of data didn't really concern me since I would also be using scrolling. This

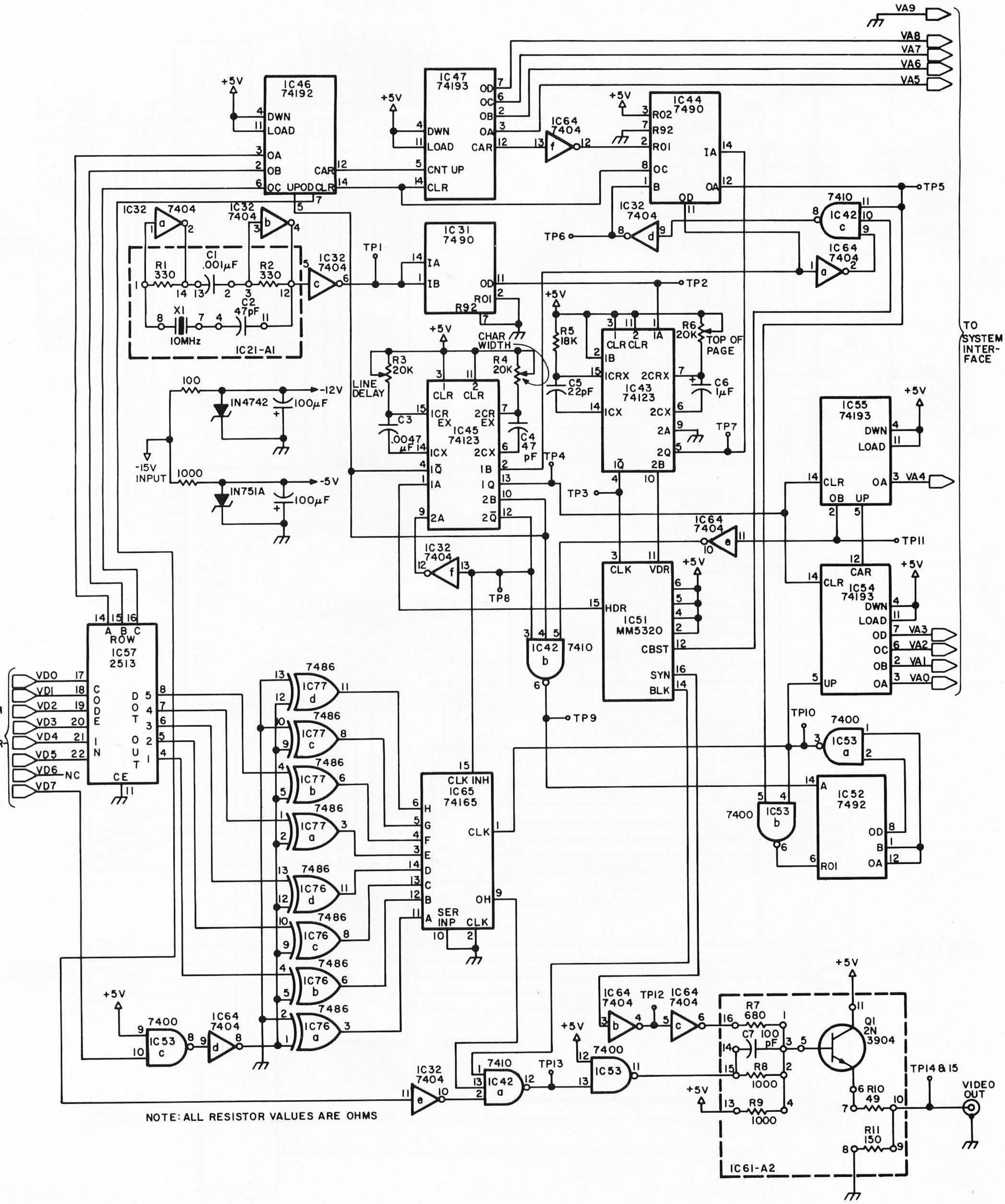


Figure 2: Detail Schematic of the Video Generation Circuitry. This section of the design includes all timing logic and character generation logic. See table 1 for a complete parts list for figures 2 and 3, and see table 2 for the power wiring list.

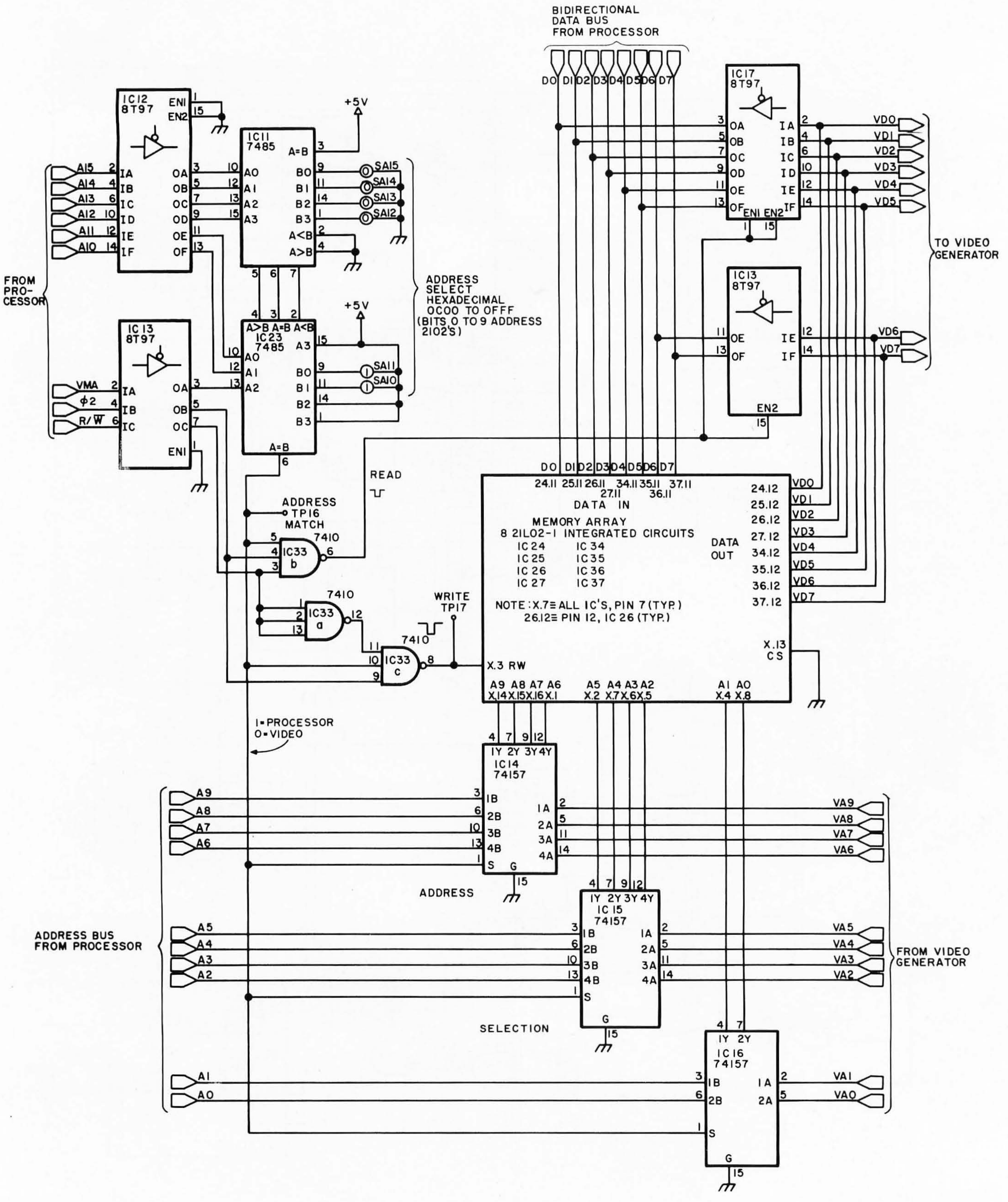


Figure 3: Detail schematic of the system interface and memory for this video terminal design. In this diagram, the processor interface is at the top and left edge, and the video generator interface to figure 2 is at the right of the drawing.

allows the display of any 16 consecutive lines of the 32 lines kept in memory. To be fancy, I thought it desirable to have both white on black characters and the reverse (black on white) for highlighting. The terminal should use the standard power supplies I had available in my system (+5, +15, −15 V). Since I didn't want to fool around modifying my television I included an RF modulator to feed the signal to the TV antenna inputs. My entire system was built on Vector #3677 and 3682 DIP plugboards. Therefore, this video terminal should also fit on the same type of board.

Finally, and most important, it had to be compatible with *MY* system. I should probably add that this terminal can of course be adapted for any computer system. It is a lot easier to interface Motorola systems to Intel systems than vice versa.

Functional Implementation

Figure 1 shows the relationship between various modules comprising this video terminal. The system incorporates a 1 K X 8 programmable memory to refresh the screen. The contents of the first 512 bytes is constantly being accessed by the video display generator. The video output is routed through coax to an RF modulator adjusted, in my case, to channel 5. This is coupled to a nine inch (22.86 cm) GE black and white portable TV through a six foot length of coax and an impedance matching transformer. The interface section of the terminal is quite simple. Its sole purpose is to allow the processor to access the video memory whenever it desires.. I felt that an occasional high priority interruption to the display was preferable to slowing down the processor. After the system was built, I discovered that no interference could be seen on the terminal when the processor was updating the display.

This display interface was built on a standard 4½" by 9½" (11.4 cm by 24.1 cm) Vector #3677 plugboard which I highly recommend. The board has two power buses etched on the reverse side. I have never experienced any problems with the power distribution while using these boards. The board was wire wrapped with all integrated circuits in sockets. Isolated discrete components were mounted with wire wrappable terminals. Analog subsystems were mounted on dual in line header plugs which, in turn, were mounted in wire wrapped integrated

Listing 1: A Memory Test Program for the 6800. This program tests the user memory of the television interface. If an error is found, the program stores the parameters of the erroneous location in locations VALUE (the test pattern to be stored), RSLT (the garbled test pattern read from the bad location) and ADDR (the address of the bad location). The limits of the test are arbitrary, being set by the constants at STRTADR and ENDADR. An error condition is concluded by entry into an endless loop of one instruction in length, thus locking up the machine until reset by the programmer making the test.

```
Addr  Hex Code  Label     Op    Operand   Commentary

0000  DE 22     MTST      LDX   STRTADR   X := STRTADR [define starting address];
0002  86 00     LOOP1     LDAA  #0        A := 0 [define first test pattern];
0004  A7 00     LOOP2     STAA  0,X       @X= A [write test pattern];
0006  01                  NOP             Let the memory circuits settle
0007  01                  NOP                down a bit between accesses;
0008  E6 00               LDAB  0,X       B := @X [read test pattern back];
000A  11                  CBA             Does test pattern read match original?
000B  26 09               BNE   ERROR     If not then quit with error;
000D  4C                  INCA            A := A + 1 [define next test pattern];
000E  4D                  TSTA            Is A = 0? [have all patterns been used?];
000F  26 F3               BNE   LOOP2     If not then continue cycle at current address;
0011  08                  INX             X := X + 1 [define next test address];
0012  9C 24               CPX   ENDADR    Is X = last address?
0014  26 EC               BNE   LOOP1     If not then restart test loops;
0016  97 1E     ERROR     STAA  VALUE     VALUE := A [indicate source value];
0018  D7 1F               STAB  RSLT      RSLT := B [indicate resulting value];
001A  DF 20               STX   ADDR      ADDR := X [indicate address of problem];
001C  20 FE     OOPSLOOP  BRA   OOPSLOOP  Hang up in endless loop until restart;
```

Data Required By MTST: These locations are at absolute addresses in the first page of memory, and are referenced by direct addressing. If the program is relocated, the variables should remain in page zero.

```
001E  00        VALUE     FCB   0         Error value;
001F  00        RSLT      FCB   0         Error result;
0020  00 00     ADDR      FDB   0         Error address;
0022  00 00     STRTADR   FDB   0         Starting address of test;
0024  00 00     ENDADR    FDB   0         Ending address of test;
```

Table 1: Video Terminal Parts List. This listing summarizes the parts identifications, sources of supply and approximate cost of this design as built by the author. The costs for most parts are typical. The only part for which it may be difficult to find alternate sources of supply is the Pixe-Verter, a brand name product. The names and addresses of the vendors used by the author are listed at the end of the table.

Quan	Type	IC number	Source	Approximate Cost*
	Integrated Circuits, TTL			
1	7400	IC53	James	.18
2	7404	IC32, IC64	James	.32
2	7410	IC33, IC42	James	.40
2	7485	IC11, IC23	James	1.78
2	7486	IC76, IC77	James	.78
2	7490	IC31, IC44	James	1.38
1	7492	IC52	James	.82
2	74123	IC43, IC45	James	2.10
3	74157	IC14, IC15, IC16	James	3.90
1	74165	IC65	James	1.10
1	74192	IC46	James	1.50
3	74193	IC47, IC54, IC55	James	4.30
	Integrated Circuits, MOS			
3	8T97B	IC12, IC13, IC17	SD	4.47
8	21L02-1	IC24, IC25, IC26, IC27, IC34, IC35, IC36, IC37	SD	12.95
1	2513N	IC57	James	11.00
1	MM5320	IC51	Nexus	4.95
	Hardware			
27	16 pin Wire Wrap Sockets		James	11.34
9	14 pin Wire Wrap Sockets		James	3.51
1	24 pin Wire Wrap Socket		James	1.05
1	14 pin DIP Plug		James	.64
1	16 pin DIP Plug		James	.70
21	Wire Wrap Terminals, type T-49		James	1.25
300′	#30 ga Wire Wrap Wire (9.95/1000′)		Ace	3.50
	Capacitors			
1	22 pF		James	.05
2	47 pF		James	.10
1	100 pF		James	.05
1	.0047 mF, Mylar		James	.12
1	.01 mF, Mylar		James	.12
31	.1 mF, ceramic bypass (3.75/50)		James	2.33
1	1 mF, Tant		James	.28
2	100 mF, 16 volt, electrolytic		James	.38
	Resistors			
1	47 ohm, ¼ watt, 5% carbon		James	.05
1	100 ohm		James	.05
1	150 ohm		James	.05
2	330 ohm		James	.10
1	680 ohm		James	.05
3	1 k		James	.15
1	18 k (particular value unit dependent)		James	.05
3	20 k, 10 Turn, Trimpots #43P203		James	4.05
	Semiconductors			
1	1N4742, 12 V Zener Diode		James	.29
1	1N751A, 5 V Zener Diode		James	.25
1	2N3904, NPN GP Transistor		James	.25
	Miscellaneous			
1	10 MHz Crystal # CY12A		James	4.95
1	Vector #3677 Plugboard		B-A	10.38
1	Vector R644 Plugboard Receptacle		B-A	4.32
1	Pixe-Verter, Model #PXV-2A		ATV Rsh	8.50
10′	RG 59/U Coax		Radio Shk	.60
1	72 ohm to 300 ohm Transformer		Radio Shk	2.95
2	RF Connectors		Radio Shk	.89
			Approximate Total Cost	**$115.28**

circuit sockets. The radio frequency modulator was located off the board in a shielded metal box underneath the computer chassis. All video and radio frequency signals are routed through 72 ohm RG 59/U coax cable. Needless to say, only good quality components were used in the construction of this system. Out of spec chips, in my opinion, are useful only for making tie tacs *[and for stapling notes to bulletin boards]*.

Hardware Configuration

Figure 2 is the schematic of the video processing subsystem. This circuitry is substantially the same as the circuit designed in the article by Mr Gantt. Several component values have been altered from his original design. 74123s were used to reduce the package count with excellent results. The dot counter clock is not derived from the crystal frequency. Instead, a one shot multivibrator is used to give variable character width. This allows one to adjust the line width to the sweep time of the individual TV being used. This frequency can be increased to allow 64 characters per line if desired. Although they are clear, I prefer not to use a magnifying glass to read the data on the nine inch TV screen. Exclusive OR gates were inserted between the 2513 character generator and the 74165 shift register to allow for reverse characters. Note that a cursor is nothing more than a reverse space. Only six data lines are required to specify a particular 6 bit ASCII subset character code. This allows the two remaining data lines to contain control information. If bit 7 is on (the most significant bit), the exclusive OR gates act as inverters to produce a reverse character. If bit 7 is off, they act as buffers, passing logically true data for a standard character. Bit 6 is currently unused. This could be used to enable a blinking gate to make individual characters flash on and off. Or it could be used to disable the character generator and enable a handblown PROM array which would allow a certain degree of graphic text capability. The video combiner

Vendors' Addresses

SD Sales; POB 28810; Dallas TX 75228.
Burstein-Applebee; 3199 Mercier St; Kansas City MO 64111.
Nexus Trading; Box 3357; San Leandro CA 94578.
James Electronics; 1021 Howard St; San Carlos CA 94070.
ATV Research; 13th and Broadway; Dakota City NE 68731.
ACE Electronic Parts; 5400 Mitchelldale B-8; Houston TX 77092.

is different from that shown in Mr Gantt's article. The 2N3904 transistor is controlled by the addition of the two inputs (composite sync and video). It provides the proper interface to the RF modulator. Note that nearly every integrated circuit is bypassed with a 0.1 mF ceramic capacitor. This insures a clean power supply line with little or no switching transient noise. I felt this to be especially important in view of the different nonrelated frequencies in the terminal. Notice the small circled numbers that are alongside of the discrete components in the crystal time base and video combiner systems. These refer to the pin locations on an IC plug which holds the components.

Anyone familiar with interface boards will recognize the simplicity of the one shown in figure 3. The address lines on the left side of the schematic are from the processor's buffered address bus. The data lines at the top are to the bidirectional 8 bit data bus. The 10 address lines and eight data lines to the right side are interconnections to the video terminal controller. The high order address lines from the processor along with the 6800's VMA (Valid Memory Address) signal are decoded by the four bit magnitude comparators IC11 and IC23. The memory address is set by the levels on the B side inputs of these 7485s. The schematic shows a setting for a memory address of hexadecimal 0C00 to 0FFF, IC23 pin 6 toggles to a positive level whenever the high order address from the processor bus matches the preset address. Within nanoseconds the data selectors (74157s, IC14, IC15, IC16) switch control of the memory address lines from the display generator to the processor bus. Half a microsecond later, with the address firmly established on the memory chips, the phase 2 clock goes high enabling the write line (if writing data) or the output buffers (if reading data). While the memory reference is being made by the processor, the terminal hardware may attempt to load a character. The character it gets is not necessarily the one at the address it wants but the character that the processor is addressing. This would seem to cause an improper character to be displayed. However, since this contention occurs so infrequently and the refresh is so rapid, no visible interference is seen on the TV screen in a typical situation. The programmable memories are of the 21L02-1 variety. Except for the data in and data out connections, the memory integrated circuits are all wired in parallel. They are permanently enabled by tying their chip select lines to ground. This hardware is about as simple as can be expected. It should not present any problems to anyone with any experience in assembly of electronics. Photos 1 to 17 show waveforms at selected test points identified on the schematics. These photos illustrate typical waveforms of my completed display board, taken from a Tektronix 455 oscilloscope.

Software Support

The primary disadvantage of this system as a terminal is the lack of a local mode like a hard copy printer or traditional display terminal. All data to be written on the screen must pass through the processor. This is not a serious disadvantage unless one desires a stand alone toy TV typewriter. You may choose to develop your own software (you have to if your processor is not a 6800!), or, if you are running a 6800 system, you might like to give mine a try.

This software is a collection of completely relocatable subroutines (that is, the following package can reside anyplace in memory without changing any subroutine addresses). There are five entry points of interest. They allow your programs to:

Table 2: Power Wiring List. This table lists each integrated circuit of this project and its power supply wiring pins. The rest of the wiring is done according to the schematics of figures 2 and 3.

No	Device	+5 V	−5 V	−12 V	GND
IC11	7485	16			8
IC12	8T97	16			8
IC13	8T97	16			8
IC14	74157	16			8
IC15	74157	16			8
IC16	74157	16			8
IC17	8T97	16			8
IC21	A1				
IC23	7485	16			8
IC24	21L02-1	10			9
IC25	21L02-1	10			9
IC26	21L02-1	10			9
IC27	21L02-1	10			9
IC31	7490	5			10
IC32	7404	14			7
IC33	7410	14			7
IC34	21L02-1	10			9
IC35	21L02-1	10			9
IC36	21L02-1	10			9
IC37	21L02-1	10			9
IC42	7410	14			7
IC43	74123	16			8
IC44	7490	5			10
IC45	74123	16			8
IC46	74192	16			8
IC47	74193	16			8
IC51	MM5320	8		1	
IC52	7492	5			10
IC53	7400	14			7
IC54	74193	16			8
IC55	74193	16			8
IC57	2513	24	12	1	10
IC61	A2	11 & 13			8
IC64	7404	14			7
IC65	74165	16			8
IC76	7486	14			7
IC77	7486	14			7

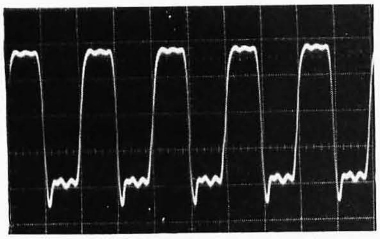


Photo 1: Test point 1, pin 1 of IC31, 50 ns per division.

Waveform Photographs

The oscilloscope photos, 1 to 17, are included for trouble shooting purposes. These were taken on a Tektronix 455 oscilloscope using a 10X probe. Unless otherwise noted, the vertical sensitivity is 1 volt per division. The horizontal sweep speed and test point location are shown in the caption of each photograph. The ground reference line is one division below the center line in each photograph.

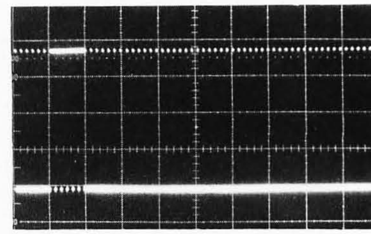


Photo 12: Test point 12, pin 4 of IC64, 200 µs per division.

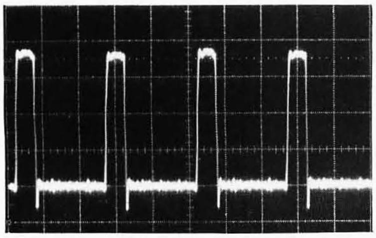


Photo 2: Test point 2, pin 1 of IC43, 200 ns per division.

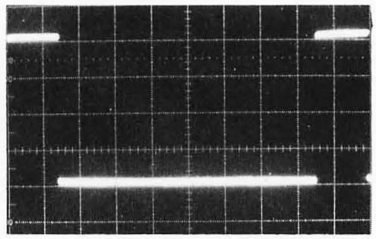


Photo 7: Test point 7, pin 5 of IC43, 2 ms per division.

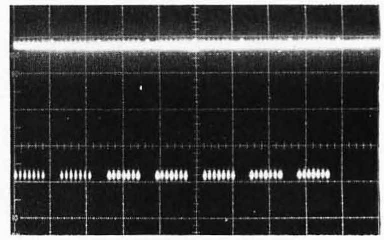


Photo 13: Test point 13, pin 12 of IC42, 500 µs per division.

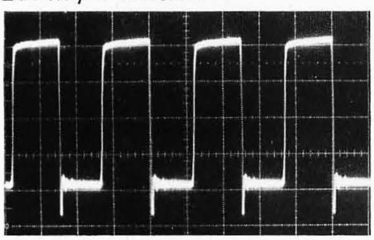


Photo 3: Test point 3, pin 4 of IC43, 200 ns per division.

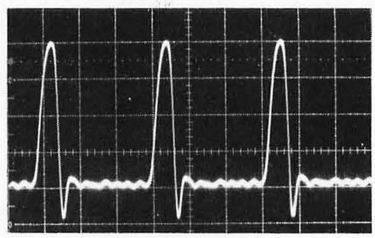


Photo 8: Test point 8, pin 12 of IC45, 50 ns per division.

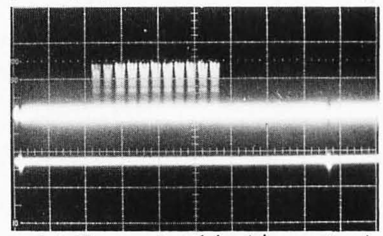


Photo 14: Test point 14, video output, 0.5 V per division vertically, 2 ms per division horizontally.

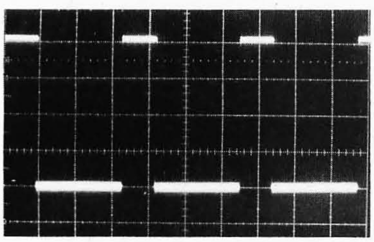


Photo 4: Test point 4, pin 13 of IC45, 20 µs per division.

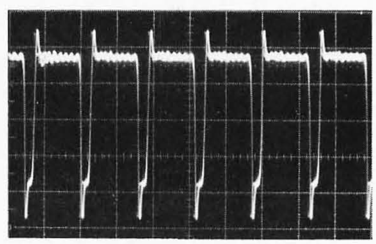


Photo 9: Test point 9, pin 6 of IC42, 100 ns per division.

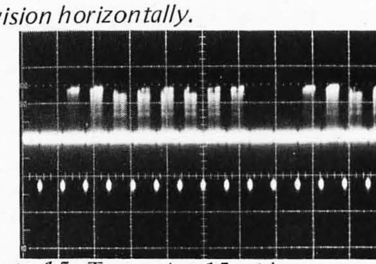


Photo 15: Test point 15, video output, 0.5 V per division vertically, 100 µs per division horizontally.

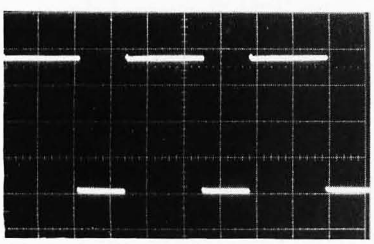


Photo 5: Test point 5, pin 12 of IC44, 5 ms per division.

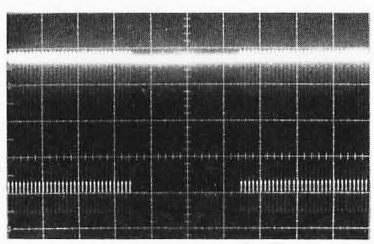


Photo 10: Test point 10, pin 3 of IC53, 10 µs per division.

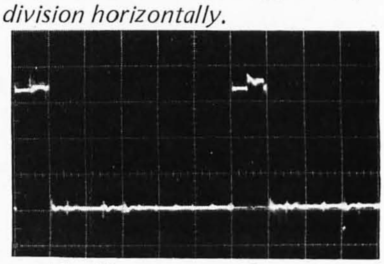


Photo 16: Test point 16, pin 6 of IC23, 1 µs per division.

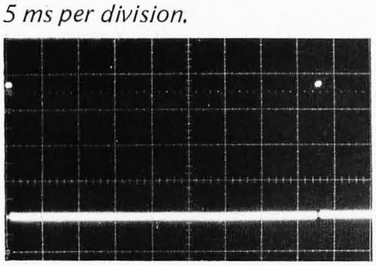


Photo 6: Test point 6, pin 1 of IC44, 2 ms per division.

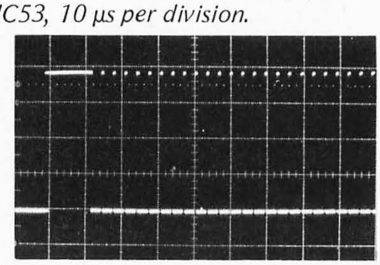


Photo 11: Test point 11, pin 2 of IC55, 200 µs per division.

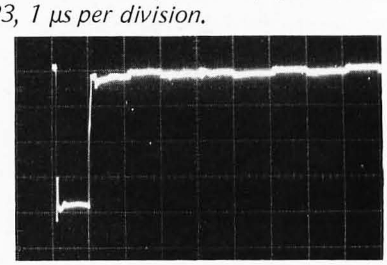


Photo 17: Test point 17, pin 6 of IC33, 500 ns per division.

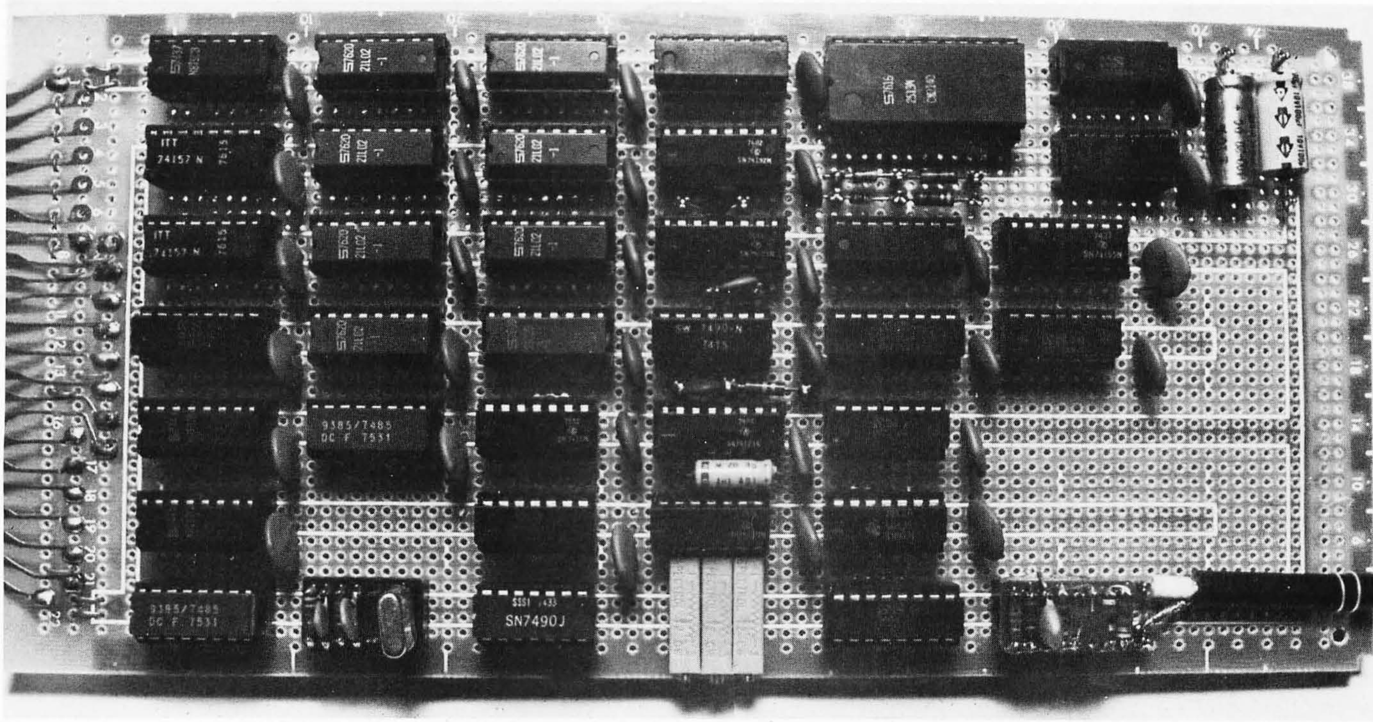


Photo 18: Video Terminal Board, Component Side. This photograph shows the physical layout of the board as constructed by the author. The processor connections to and from the terminal are made by the plug on the left hand side of the photograph. Eight 21L02-1 memory integrated circuits are used for the screen's local memory. The three pots at the bottom center are used (L to R) to control left hand margin, character width, and top of page margin. The video amplifier is contained in the DIP plug with components, in the lower right hand corner. Bypass capacitors are sprinkled quite liberally throughout the board.

1. Display a new line of ASCII text.
2. Scroll all lines up by one.
3. Erase one character line.
4. Add one new character to the display.
5. Erase the entire screen.

The preambles for each subroutine should adequately explain the calling sequence and conventions used. These subroutines assume that six words are available in the first page (256) of memory (hexadecimal address 0 to FF). These words are used to take advantage of Motorola's direct addressing feature. To display a character in a preset location on the screen, initialize the words CHAR and LINE to the desired location, load the 6 bit ASCII character into the A accumulator and branch to the VCHAR subroutine.

The complete relocatable software is included as listing 2 with this article. A symbolic example is given here to show the use of these routines. Suppose that you desire to erase the screen, display a title, a prompting message, and place a cursor at the next character location. This can be accomplished by the following code:

```
START    JSR    ESCRN      Erase the entire screen;
         LDX    #TITLE     X := address of title string;
         JSR    VLINE      Write title at top of screen;
         LDX    #PROMPT    X := address of prompting message;
         JSR    VLINE      Write prompting message on next line;
         LDAA   #$A0       A := ' ' + 80 [add hex 80 to blank giving inverted blank];
         JSR    VCHAR      Write cursor at location of next input;
         .
         .
         .
TITLE    FCC    'GENIE OPERATING SYSTEM'          Title string;
         FCB    4
PROMPT   FCC    'WHAT IS YOUR DESIRE, MASTER?'    Prompting string;
         FCB    4
```

Or if you really have to have a routine which will allow you to type data on your keyboard and look at it on the screen:

```
LOCAL    JSR    ESCRN      Start by clearing the screen;
LOOP     JSR    INCH       Use normal input routine to define character in A;
         JSR    VCHAR      Transfer input character to screen buffer;
         CMPA   #$04       Is it end of text?
         BNE    LOOP       If not then continue local LOOP;
         .
         .
         .
         Return to monitor
```

Note that scrolling is done automatically for you after you write the 32nd character on the 16th line.

Construction

Before you can begin building this terminal, you first must obtain the necessary components. Refer to the component list in table 1. I have listed the vendors from which

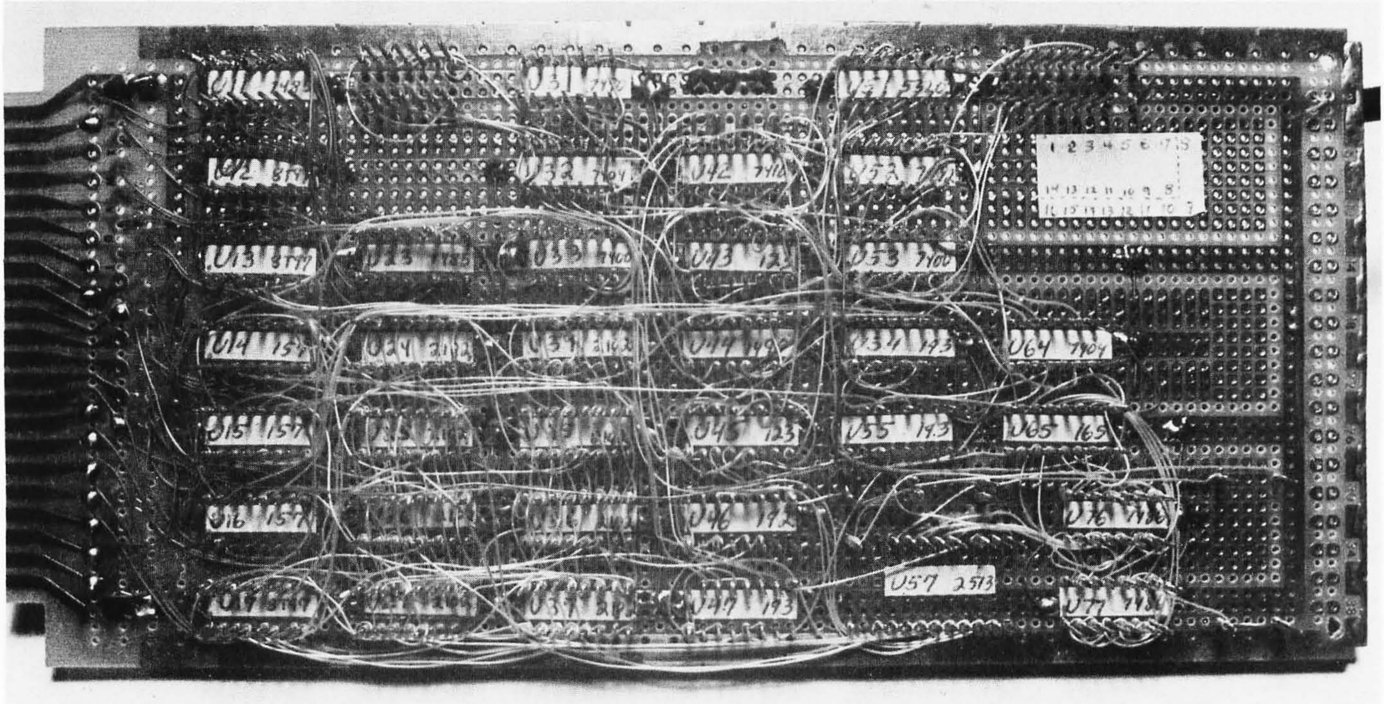


Photo 19: Video Terminal Board, Wiring Side. This photograph shows the wire wrapped construction of the circuit's prototype. All integrated circuit locations were labelled with masking tape containing device location and device type. All integrated circuit power connections were soldered to the etched power supply buses which are part of the Vector plugboard used for the circuit. The orientation of pin 1 is uniformly the upper left corner of the integrated circuit, as documented by the typical pinout map shown pasted to the upper right side of the board.

I purchased the parts. I selected these people because I knew I would get quality components at reasonable prices in a short time. The prices listed are included as a guide. Check the advertisements in this issue to see if they have changed.

The first step in building this (or any) board is to determine exactly where the various components will be located. See photo 18, photo 19 and figure 4 for the layout I used. Then install all of your bypass capacitors. This is easy to do at this stage. Solder them directly to the etched lands on the reverse of the Vector board between the known locations of the wire wrap sockets. Next (and this is a trick that will really reduce the wire wrap errors) cut masking tape into sections a little smaller than the size of the integrated circuit top. Write on the masking tape the integrated circuit number and the TTL suffix. For example,

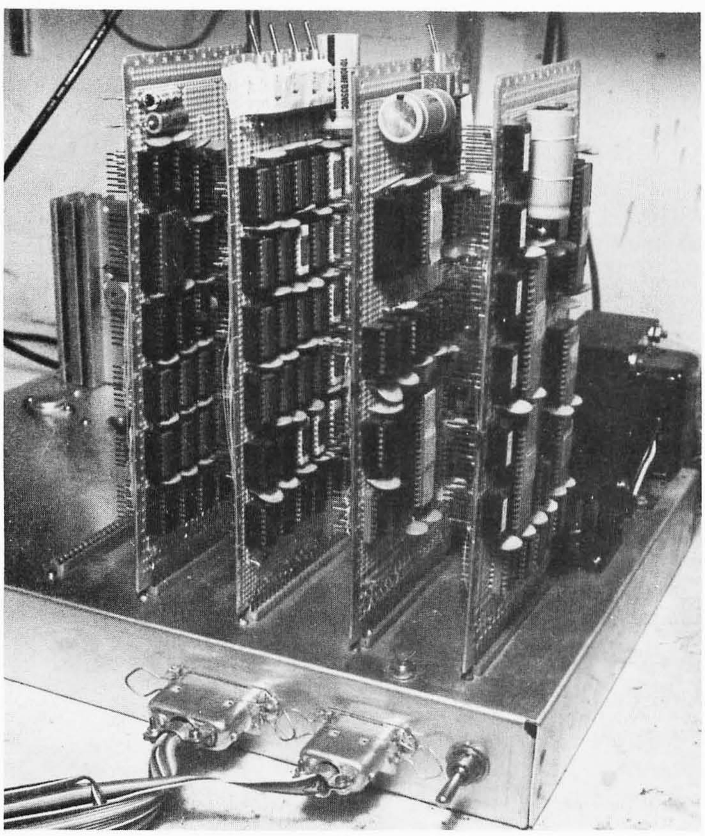


Photo 20: A Detail of the Computer System. This 6800 microcomputer currently consists of four cards: an IO card, a processor card, a 4 K byte programmable memory card, and the video terminal card described in this article. One spare location at the left is awaiting a 16 K dynamic programmable memory card which is under development. The ribbon connectors in the lower right hand corner are used to interface a control panel. Each board is supplied from a separate 5 V regulator which is mounted on the chassis for superior heat sinking capability.

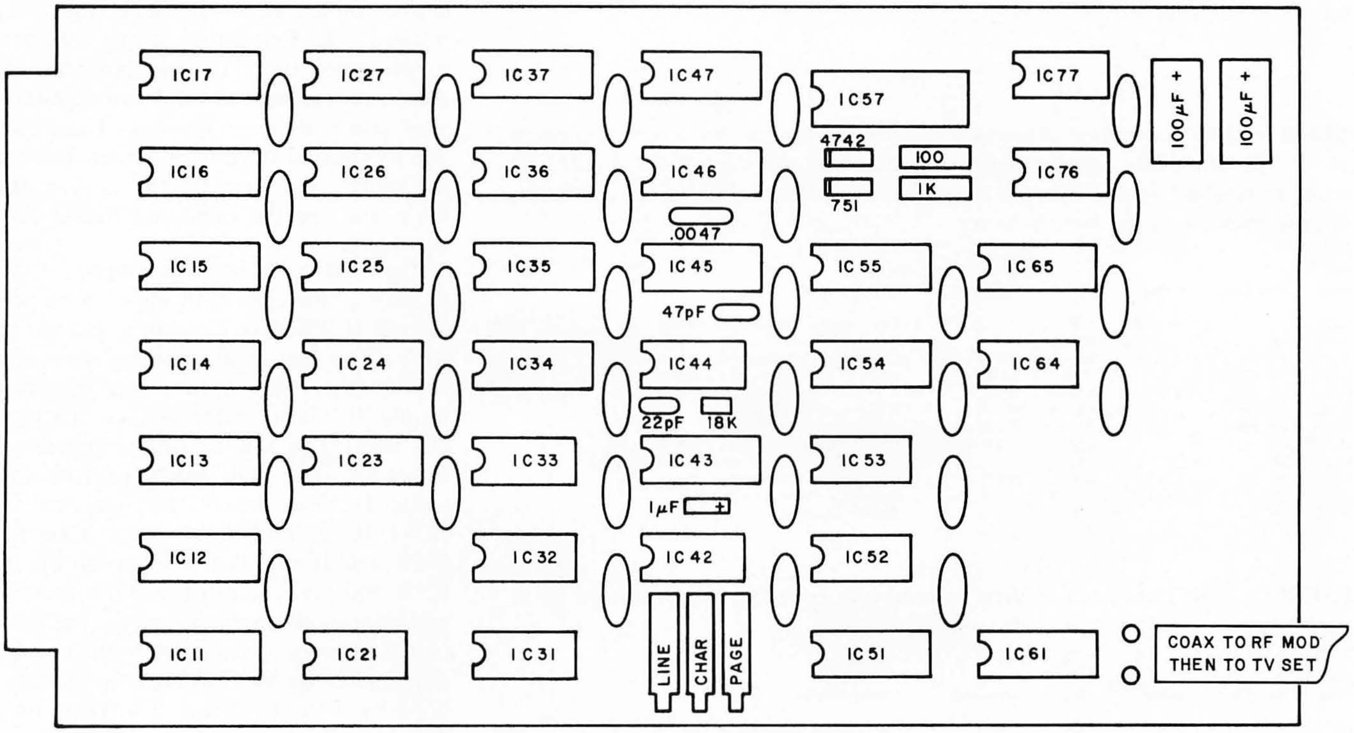


Figure 4: Video Terminal Component Layout. This reduced scale drawing shows the relative position of the parts of the video terminal design as constructed by the author. See photo 18 for a pictorial version of this layout, and see photo 19 for the reverse side of the author's card as wired for his system.

"IC45-123" identifies the fifth integrated circuit in the fourth column as a 74123. Pin 1 is *always* in the upper left hand corner looking from the wrap side of the board. Place the label on the wrap side of the board where the socket will go. Next, put a dab of cement on the bottom of the integrated circuit socket and insert it on the board. (I used "DUCO" cement. Epoxy or hot melt glue can be used, but are nearly impossible to remove once set.) Remember that 16 pin sockets are a little longer than the 14 pin variety and you'll be OK. Refer to figure 4 for the configuration I used. After the glue has set, wire the power supply lines to the wire wrap sockets. To do this, wrap half a dozen turns around a power pin and solder the other end to the nearest power bus. Don't "daisy chain" a bunch of circuits together with one attachment to the power bus. The 30 gauge wire can't adequately handle the required switching current for more than a few chips. Insert the wire wrap terminals where the discrete components will be placed. Many of these can be soldered directly to the power planes. Otherwise, they have enough friction to prevent them from falling out before they are wrapped. It's time for one more trick to reduce the number of wire wrap errors. Get a Xerox copy of the schematics (figures 2 and 3). *[BYTE encourages this and gives full permission for such private use duplication.]* Each time you complete a connection, use a felt tipped pen to draw over the line on the

Listing 2: Complete Video Support Software for a Motorola 6800 Processor. This package consists of self relocating routines to perform display utility functions. Self relocation is accomplished by exclusively employing branch instructions (relative addressing) for program control within the package, and using the stack as a temporary where necessary. Control variables are assumed to be in low memory.

The following allocations are located in page 0 of memory, so that the video support software can take advantage of the Motorola 6800 direct addressing mode. Relocation of these variables should be confined to page 0, in which case the address constants used for the direct addressing references would be altered to reflect new locations. The main text of the subroutines themselves is completely position independent, which means relocation can be accomplished simply by loading the program code at a different point in memory address space and calculating new addresses for the "external references" summarized in a table at the end of this listing.

Abs Addr	Hex Code	Label	Op	Operand	Commentary
0020	00	CHAR	RMB		Character position, 0 (left) to 31 (right);
0021	00	LINE	RMB		Line number, 0 (top) to 15 (bottom);
0022	00 00	CHPSN	RDB		Memory address pointer to character;
0023		CHPSNL	EQU	CHPSN+1	Low order byte of memory address pointer;
0024	00 00	SAVEX	RDB		Index register save area;

The following equate assigns a value to the address of the video display memory for the purposes of symbolic reference within this listing. The addresses of the displayed portion of memory in the prototype of this design are 0C00 to 0FFF in hexadecimal.

```
VIDEO    EQU    =$0C00
```

Listing 2 is continued on the next three pages . . .

Listing 2, continued:

VLINE — *This subroutine displays a new line of text on the screen. On entry to VLINE, the index register X has the address of the start of the ASCII text string to be.displayed. The string is terminated using an ASCII EOT character, hexadecimal 04, as the last character.*

Rel Addr	Hex Code	Label	Op	Operand	Commentary
0000		VLINE	EQU	*	Entry point, line display routine;
0000	36		PSHA		Save A on stack;
0001	7F 00 20		CLR	CHAR	CHAR := 0 [perform carriage return] ;
0004	7C 00 21		INC	LINE	LINE := LINE + 1 [perform line feed] ;
0007	A6 00	VLP	LDAA	0,X	A := @X [move next character to A] ;
0009	81 04		CMPA	4	Is it end of text?
000B	27 05		BEQ	VLDNE	If so then return to caller;
000D	8D 15		BSR	VCHAR	CALL VCHAR [display the character] ;
000F	08		INX		X := X + 1 [point to next character] ;
0010	20 F5		BRA	VLP	Reiterate until done;
0012	32	VLDNE	PULA		Restore A from stack;
0013	39		RTS		Return to caller;

ESCRN — *This subroutine is used to erase the entire screen. It has no input parameters.*

Rel Addr	Hex Code	Label	Op	Operand	Commentary
0014		ESCRN	EQU	*	Entry point, erase screen routine;
0014	36		PSHA		Save A on stack;
0015	86 0F		LDAA	#15	A := 15 [set number of lines to erase] ;
0017	8D 6B	ESLOP	BSR	ELINE	CALL ELINE **[erase one line] ;**
0019	4A		DECA		A := A − 1 [decrement line count in A] ;
001A	2C FB		BGE	ESLOP	if A >= 0 then continue the loop;
001C	7F 00 21		CLR	LINE	LINE := 0 [move cursor to top of screen] ;
001F	7F 00 20		CLR	CHAR	CHAR := 0 [move cursor to left of line] ;
0022	32		PULA		Restore A from stack;
0023	39		RTS		Return to caller;

VCHAR — *This subroutine displays one character on the screen. On entry, the A accumulator should contain the ASCII character code to be displayed. VCHAR updates the control variables LINE and CHAR which determine screen location; VCHAR ignores all ASCII control codes except a carriage return or line feed.*

Rel Addr	Hex Code	Label	Op	Operand	Commentary
0024		VCHAR	EQU	*	Entry point of character display routine;
0024	81 0D		CMPA	#$0D	Is character a carriage return?
0026	27 09		BEQ	VCR	If so then go juggle CHAR pointer;
0028	81 0A		CMPA	#$0A	Is character a line feed?
002A	27 09		BEQ	VLF	**If so then go juggle LINE pointers;**
002C	81 20		CMPA	#' '	**Check for ASCII control character;**
002E	2C 09		BGE	VOK	If so then go transfer valid ASCII;
0030	39		RTS		Else return with no action;
0031	7F 00 20	VCR	CLR	CHAR	CHAR := 0 [perform carriage return] ;
0034	39		RTS		Return to caller;
0035	7C 00 21	VLF	INC	LINE	LINE := LINE + 1 [perform line feed] ;
0038	39		RTS		Return to caller;
0039	37	VOK	PSHB		Save B on stack;
003A	36		PSHA		Save A on stack;
003B	D6 20		LDAB	CHAR	B := CHAR;
003D	C1 1F		CMPB	#31	Is it still on the same line?
003F	2F 06		BLE	VCOK	If so then go test line [do not pass CRLF] ;
0041	7F 00 20		CLR	CHAR	CHAR := 0 [perform carriage return] ;
0044	7C 00 21		INC	LINE	LINE := LINE + 1 [perform line feed] ;
0047	96 21	VCOK	LDAA	LINE	Is LINE on
0049	80 0F		SUBA	#15	the screen?
004B	2F 0B		BLE	VLOK	If so then go to display [do not pass home] ;
004D	8D 1D	VLNOK	BSR	VSCROL	CALL VSCROL [move all lines up one] ;
004F	4A		DECA		A := A − 1 [decrement scrolling count] ;
0050	26 FB		BNE	VLNOK	if A >= 0 then loop [until line on screen] ;
0052	86 0F		LDAA	#15	A := 15 [position cursor at bottom line] ;
0054	97 21		STAA	LINE	
0056	8D 2C		BSR	ELINE	CALL ELINE [erase the bottom line] ;
0058	96 21	VLOK	LDAA	LINE	Set up LINE parameter for GCHAR call;
005A	D6 20		LDAB	CHAR	Set up CHAR parameter for GCHAR call;
005C	8D 3E		BSR	GCHAR	CALL GCHAR [to calculate CHPSN pointer] ;
005E	DF 24		STX	SAVEX	Save index register value;
0060	DE 22		LDX	CHPSN	X := CHPSN [point to current character address] ;
0062	32		PULA		A := current character;
0063	A7 00		STAA	0,X	@X := A [store current character in memory @ X] ;
0065	7C 00 20		INC	CHAR	CHAR := CHAR + 1 [set pointer for next write] ;
0068	DE 24		LDX	SAVEX	restore old index register value;
006A	33		PULB		restore B from stack;
006B	39		RTS		return to caller;

copy. You are done when all of the lines are marked over. Remember to mark the power supply lines too. I should probably insist that you recheck all of your connections after you're done . . . however, I don't and I don't expect that you will either. It is easier to verify your connections as you make them and fix the ones you missed during checkout.

Now that your board is wrapped it's time to stuff all the circuits in and turn the power on, right? WRONG. Go slowly and test your work. You may end up saving some expensive circuits. First, measure your +5 V power supply. If it is not within 0.25 V of 5 V, find out why. Too low a voltage impedes performance. Too high and your chip life is reduced. Next, install the interface chips (IC11, IC12, IC13, IC14, IC15, IC16, IC17, IC23 and IC33). Put a scope on pin 1 of IC16 and verify a positive pulse each time you address this memory space. The address of the memory space is determined by the B side inputs to the 7485s. This test can be done by a front panel or a software loader. This select pulse should be about 1.0 μs in duration if you are running with a 1 MHz clock. Verify the address inputs at the programmable memory socket locations to make sure the 74157s are switching correctly. Look at IC33 pin 3 and see a 500 ns negative pulse (normally high, goes low for about 500 ns, then high again) when you are writing to the programmable memory. Verify that there is no pulse when reading. Check for the reverse at IC33 pin 6. Expect a negative pulse when reading and just a high level when writing. Turn the power off and insert the eight 21L02-1 memory circuits. Each time a circuit is inserted, look to make sure all the pins are in the socket and not bent over. Power up again and run your memory test program. This verifies that your system can access this memory and that the integrated circuits are all good. An example of a simple memory test program called MTST is included in this article as listing 1.

When you are sure that the processor is interfaced to memory correctly, insert IC32, IC31, and IC43. Mount the DIP plug with the oscillator components in the IC21 socket. Look at IC31 pin 1 on your scope and check for a 10 MHz "square" wave. See photo 1 for an example. Photo 2 shows the 2 MHz wave at IC43 pin 1. If all is well so far, put your probe on IC43 pin 4. This should be pretty close to a 2 MHz square wave. Adjust the timing resistor, if necessary, to obtain a 50% duty cycle. This doesn't have to be exact but the 5320 is happier with a square wave. Measure pin 1 of IC51 and verify a −12 V (±5%) DC voltage. With power off, insert this magical inte-

grated circuit into its socket. This 5320 generates all sorts of good signals and saves a lot of board space. With power on (and pin 1 still at a *clean* −12 V), scope pins 11, 12, 14, 15, and 16. They will be switching between +4.5 and −11 V. Don't be alarmed by the negative output voltage. They will pop up to ground when a TTL load is hung on them. With power off again, insert IC45, IC46, IC47, IC64, IC44, and IC42. Adjust the Line Delay pot for a 20 μs positive pulse at IC55 pin 14. This adjusts the left hand margin on the TV display. It will be trimmed to a final value at a later time. Adjust the Top of Page pot for a 3 ms positive pulse at IC44 pin 14 (this will also be trimmed later). Verify pulses at IC57 pins 14, 15, and 16. Set the Character Width pot to midrange (about 6.25 MHz at IC32 pin 12). Refer to the waveform photographs if problems are encountered. Insert everything else except for the character generator (IC57). Check that the address lines are toggling and that data is being presented to the character generator on pins 17 through 22. IC65 pin 9 should be switching erratically. Verify the power supply connections to IC57. The +5, −5, and −12 V connections should be within 5% and should be *clean*, that is, without more than 100 mV of noise.

Turn the power off and remove the 2513 from its protective packing. This is the last device to be added to the board. It is also the most expensive circuit so don't blow it. With power on, recheck the supplies and the outputs from this circuit.

Connect the video output from this terminal module to the video input of an RF modulator. I obtained excellent results using a Pixe-Verter. The RF output is transmitted by coax to a 72 to 300 ohm transformer. Disconnect the antenna from the back of your TV and attach the leads from this transformer. Tune to the channel you selected when building the Pixe-Verter with the TV's fine tuning set to midrange. With power on adjust the tuning capacitor on the Pixe-Verter for some sort of display on the TV. It will look ridiculous at first. You'll have to readjust the horizontal and vertical hold controls on your TV for a stable display. Trim the three pots for a pleasing display on the TV screen. Then you can go to town using the display.

The results of my project are shown in photos 20 and 21.

Trouble Shooting

Fortunately, this version of a video terminal worked the first time it was tried. This means I don't have any suggestions for locating failures based on experience. If you run into trouble, compare your waveforms

Listing 2, continued:

VSCROL — *This subroutine moves all lines on the display up by one position. No input parameters are necessary.*

Rel Addr	Hex Code	Label	Op	Operand	Commentary
006C		VSCROL	EQU	*	Entry point of scrolling subroutine;
006C	36		PSHA		Save A on stack;
006D	DF 24		STX	SAVEX	Save index register value;
006F	CE 0C 00		LDX	#VIDEO	Load pointer for start of move;
0072	A6 20	VSCRL	LDAA	32,X	Fetch corresponding character, next line;
0074	A7 00		STAA	0,X	Then store it in the current line;
0076	08		INX		X := X + 1 [point to the next character];
0077	8C 0F E1		CPX	#(VIDEO+32*(32−1)+1)	[calculated start of the last line in a 1024 byte buffer];
007A	26 F6		BNE	VSCRL	if not at end of buffer, then repeat loop;
007C	86 0F		LDAA	#15	line parameter of erase function;
007E	DE 24		LDX	SAVEX	restore index register
0080	8D 02		BSR	ELINE	CALL ELINE [to erase bottom line];
0082	32		PULA		Restore A from stack;
0083	39		RTS		Return to caller;

ELINE — *This subroutine is used to erase one line from the display. The line number is specified by the value in accumulator A upon entry.*

Rel Addr	Hex Code	Label	Op	Operand	Commentary
0084		ELINE	EQU	*	Entry point of line erase routine;
0084	37		PSHB		Save B on stack;
0085	36		PSHA		Save A on stack;
0086	5F		CLRB		B := 0 [set local character position to zero];
0087	8D 13		BSR	GCHAR	CALL GCHAR [calculate character address];
0089	DF 24		STX	SAVEX	Save index register;
008B	DE 22		LDX	CHPSN	X := CHPSN [get calculated character address];
008D	C6 20		LDAB	#32	B := 32 [the number of characters in a line];
008F	86 20		LDAA	#' '	A := ' ' [erase means put a blank in each pos.];
0091	A7 00	ELLOP	STAA	0,X	@X := A [move the blank to current location];
0093	08		INX		X := X + 1 [point to next location];
0094	5A		DECB		B := B − 1 [decrement loop counter];
0095	26 FA		BNE	ELLOP	If count remains then reiterate;
0097	DE 24		LDX	SAVEX	Restore index register;
0099	32		PULA		Restore A from stack;
009A	33		PULB		Restore B from stack;
009B	39		RTS		Return to caller;

GCHAR — *This subroutine calculates the memory address of a character specified by the current line number, LINE, and the current character position, CHAR. The formula is:*

*CHPSN := 32 * A + B + BASEADDR ;*

Where A is the line number, passed in accumulator A, B is the character position value, passed in accumulator B, and BASEADDR is a constant giving the first address in the character display area of memory. The value of BASEADDR in the author's system is hexadecimal 0C00. Since the low order offset is zero, this constant is only added into the high order at location 00C0 of this listing.

Rel Addr	Hex Code	Label	Op	Operand	Commentary
009C		GCHAR	EQU	*	Entry point of the address calculator;
009C	36		PSHA		Save A on stack;
009D	37		PSHB		Save B on stack;
009E	5F		CLRB		Clear the character
009F	D7 22		STAB	CHPSN	address variable
00A1	D7 23		STAB	CHPSNL	prior to calculation;
00A3	4A	GLOP	DECA		A := A − 1 [decrement line count];
00A4	2D 0A		BLT	GCDS	If A = 0 then multiplication is done;
00A6	CB 20		ADDB	#32	A := A + 32 [multiply is repeated addition];
00A8	24 F9		BCC	GLOP	If no carry then continue inner loop;
00AA	7C 00 22		INC	CHPSN	Else increment most significant byte,
00AD	0C		CLC		clear the carry flag,
00AE	20 F3		BRA	GLOP	and then return to inner loop;
00B0	D7 23	GCDS	STAB	CHPSNL	CHPSN+1 := A [save the low order byte];
00B2	33		PULB		Restore the displacement in line from B;
00B3	17		TBA		A := B [put it in A for calculation];
00B4	9B 23		ADDA	CHPSNL	A := A + CHPSNL [add low order bytes];
00B6	97 23		STAA	CHPSNL	CHPSNL := A [and save result in pointer];
00B8	24 04		BCC	GDONE	If no carry, then calculation is completed;
00BA	7C 00 24		INC	CHPSN	Else propagate carry to high order byte,
00BD	0C		CLC		and clear carry;
00BE	96 22	GDONE	LDAA	CHPSN	A := CHPSN [fetch high order byte];
00C0	8B 0C		ADDA	#H(VIDEO)	A := A + high order of VIDEO address;
00C2	97 22		STAA	CHPSN	CHPSN := A [save final high order byte];
00C4	32		PULA		Restore A;
00C5	39		RTS		Return to caller;

Listing 2, continued:

Notation conventions:

RMB	= "reserve memory byte" with or without label, uninitialized.
RDB	= "reserve double byte" with or without label, uninitialized.
EQU	= "equate". Assign an address value to a label.
#	=• indicator for immediate addressing on Motorola 6800.
$nnn	= hexadecimal value nnn.
nnn	= decimal value nnn.
*	= the current location counter as in an assembly.
'X'	= notation for ASCII character X.
H(VIDEO)	= notation for high order byte of address VIDEO.
@X	= data at location X (used in commentary).

External Symbol Table:

The following symbols and their relative addresses are the entry points to the video display support software. To calculate the absolute addresses needed to reference these entries, add the starting address at which you load the package to each of these offsets, giving the absolute address to reference in a jump to subroutine instruction.

Relative Address	Label	Operation
0000	VLINE	Line display
0014	ESCRN	Erase screen
0024	VCHAR	Display one character
006C	VSCROL	Scroll up one line
0084	ELINE	Erase one line
009C	GCHAR	Calculate address given line, position

with those shown in the photos 1 to 17. The most likely cause of problems are defective integrated circuits and incorrect wiring.

Future Additions

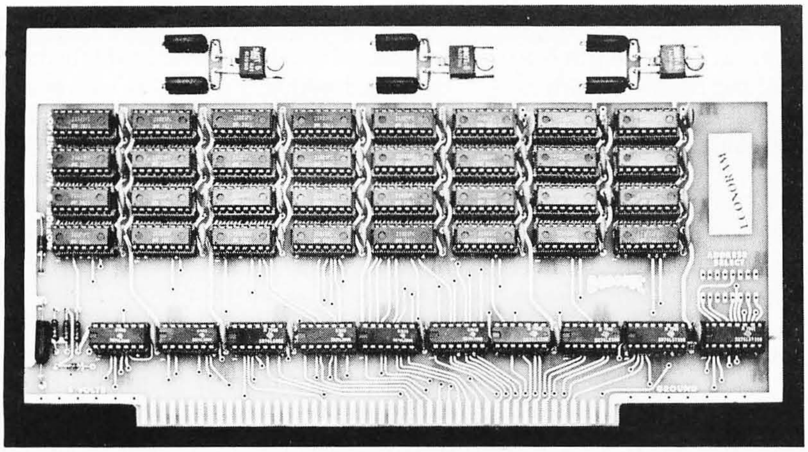
If you want to display more characters on the TV screen, try disconnecting the wire from IC55 pin 2 to IC64 pin 11. Replace with a wire from IC55 pin 6 to IC64 pin 11). Wire IC55 pin 2 to the 9th address bit at IC14 pin 2. Readjust the Character Width pot to squeeze all those characters on a single line. The software will have to be adjusted to handle the new memory configuration of 16 by 64.

Like to try graphic text? Functionally replace the 2513 with your own PROM. Bit 6 could be used to perform this selection electronically. This PROM could be used to define various line segments within a character field depending upon the 6 (or 7) bit code supplied.

The very nicest thing about a homebuilt project is that after it is working, you're not afraid to modify it! Happy Wrapping.■

Photo 21: A Microcomputer Development Workshop? The processor is off in the left hand corner of the table, with a cassette tape recorder to its right. A keyboard with ASCII encoding is connected to the IO card for control of the processor. The video display's television set output is to the left of the oscilloscope at the right side of the picture. The parts cabinets, wire wrap gun, soda pop and ash tray were essential to the development of this system.





We took everything we learned from selling 4K x 8 RAM boards for the past year, added some of this year's circuit tricks, and came up with ECONORAM --- a memory board that is even more remarkable due to its low price. We've engineered this with the user in mind, giving you several benefits:

* 3 regulators to share power load, plus optimized thermal design, means a cooler running microcomputer
* Typical current consumption of under 750 ma gives your power supply a break
* Fast --- Zero wait states
* All TTL support ICs are latest Low Power Schottky types
* For reliable and unambiguous data transfer, all addresses, data lines, and outputs are buffered for minimum loading and maximum output capability
* Power-on clear included

All these features are packed on to an Altair-sized, industrial quality double-sided PC board, with sockets for all ICs, 7 tantalum capacitors for power supply decoupling, and plenty of bypass capacitors---39 of them, in fact, as well as a logic print and instructions. ASSEMBLED/TESTED UNITS ALSO AVAILABLE FOR A SURCHARGE; write for quotations and specify quantity.

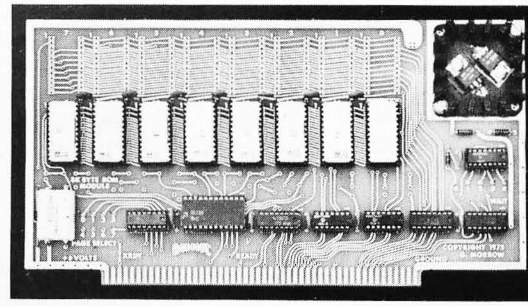
ECONORAM™

$99.95

+2 LBS POST

Our popular **ECONORAM 4K x 8 RAM** board is now available **assembled, tested, and warranteed for one year.** Plug it in to your Altair or IMSAI and enjoy the same performance that has made the kit such a success---guaranteed zero wait states and current drain of 750 mA or less; on board regulation, easy address selection, and lots more.

NOW! assembled ECONORAM $129.95



"EconoRom" $179.95

ALTAIR 8800/IMSAI PLUG-IN COMPATIBLE. This is a 4K by 8 EROM board...the ideal place for putting software, be it assembler, editor, or any custom set of routines. Additionally, this board may expand to 8K x 8 by simply adding more sockets and EROMs; also available is a 2K x 8 version if you don't need a full 4K. LOW POWER: 8K board requires ½A @ 5V, & 150 ma @ -12V. Buffered addresses for lightest loading, buffered outputs for maximum drive. Kit includes sockets, double-sided quality PC board, on board regulators, logic print, and instructions. Program it yourself, or have us do the programing.

8K X 8 BOARD $269.95   2K X 8 BOARD $135.00

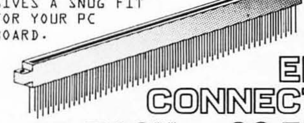
8080 Software Board $189.95

We took our ECONOROM board kit, but instead of including blank EROMs, these are programmed with assembler, editor, & monitor routines for the 8080. This is a valuable first step if you're trying to get away from machine language programing. There's not really enough room here to fully describe all the functions of the software...but if you send us $2.95 (refundable with order), we'll send you our software packet that includes instructions listing, schematic, and assembly data.

74LS TTL

00	$0.36	30	$0.36	132	$1.50	175	$1.35
01	0.36	32	0.38	138	1.38	221	1.38
02	0.36	37	0.53	139	1.38	240	1.88
04	0.42	38	0.53	155	1.38	257	1.25
08	0.38	42	1.25	157	1.25	258	1.38
10	0.36	74	0.56	160, 161,		273	2.25
11	0.38	75	0.85	162, 163=		283	1.20
20	0.36	109	0.60	$1.85 ea.		367	1.00
21	0.38	124	2.50	168	1.87	368	1.00
22	0.38	125	0.75	169	1.87	377	1.88
27	0.38	126	0.75	174	1.38	378	1.38

GOLD-PLATED 3 LEVEL WIRE WRAPPING POSTS. IDEAL FOR MATING WITH ALTAIR/IMSAI PERIPHERALS; GIVES A SNUG FIT FOR YOUR PC BOARD.



100 PIN EDGE CONNECTOR

$5 EACH--$22 FOR 5

IMSAI/ALTAIR EXTENDER BOARD KIT

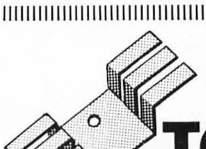
--BUILT IN LOGIC PROBE $35.00
--ALTAIR/IMSAI COMPATIBLE
--ALLOWS YOU TO TROUBLESHOOT AND WORK ON BOARDS OUTSIDE OF THE SYSTEM. IDEAL FOR TAKING MEASUREMENTS; MAKES PROBING THE BOARD EASY.

LIMITED QUANTITY SPECIALS!

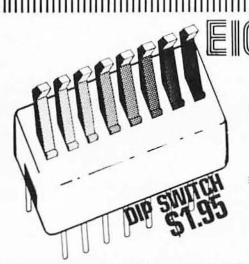
9601 Precision one-shot.................3/$1.00
74H40 High speed 7440...................7/$1.00
MM5262 2K dynamic RAM. Many bits per buck!..$1.25
FPQ3725 Quad PNP transistor array, type 2N3725, in a DIP package. Get 4 FPQ3725s for...$1.95

More ICs

74S471 8 x 256 fast ROM in DIP package.......$9.50
81LS95/96/97/98 Tri-state LS octal buffers..$1.13
2102L1 **Low power**, 450 ns guaranteed for full temp range. Just in.....................$1.95



4/$1 TO-220 HEAT SINKS



EIGHT POLE SINGLE THROW TOGGLE ACTION

TERMS: Orders under $10 add 50¢ handling. No COD orders. Place Mastercharge® and BankAmericard® orders by calling **415-562-0636**, 24 hours. California residents add appropriate sales tax.

GODBOUT

BILL GODBOUT ELECTRONICS
BOX 2355, OAKLAND AIRPORT, CA 94614

While you're here...circle the reader service card. Our flyer will make it worth your while!

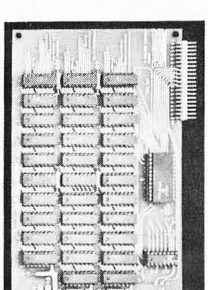
GROUP DISCOUNTS: COMPUTER CLUB MEMBERS MAY WISH TO POOL THEIR ORDERS. BUY 10 KITS, TAKE 10%; BUY 100, TAKE 20%.

4Kx8 NAKED RAM $88

This 4K by 8 memory board has no frills, just storage. Designed for compatibility with JOLT systems, this board is also ideal for other microcomputers using bi-directional buss systems. Same size as JOLT memory card, plus low-power operation (750 ma) to keep you on good terms with your power supply.

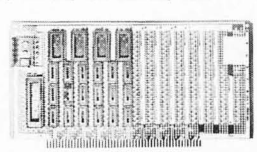
If you don't need the onboard regulation or address / data buffers of our "Bigger Brother" 4K board, then this is the way to go.

Sockets included for all ICs.



from VECTOR!

This is one of the neatest Altair accessories we've seen. It accepts virtually any size IC package, has a power and ground plane on opposite sides of the board for extra capacitance. Room for 4 regulators, 1 heat sink provided with board. By the way, the sockets are shown only for illustration, but they get the point across that you can stuff a lot of ICs on here --- implement your own memory boards, I/O boards, etc.



UNIVERSAL PROTOTYPE BOARD ➡ ➡ ➡ $ 19.95

PACE PACKET $2.50

Learn about PACE, the 16 bit microprocessor chip, with this 80 page booklet. Contains detailed and specific information on the chip itself, software, system organization, and more. Also includes 11" X 17" fold-out logic print of PACE implementation.

MicroMONITOR $29.95

We've taken the 5204, a 4K EROM, and programmed it with some software for your 8080. With this software, you may:
**Examine and fill memory
**Punch and read paper tape
**Branch to, and recover from, programs.

Our package consists of the programmed 5204 as well as a complete software listing.

Continued from page 89

Many people have done as I have, which is to put aside their 8008 and replace it with an 8080A or newer generation equipment. This 8008 unit which is kicking around can solve the chief disadvantage of vector graphics – software refresh. The 8008 is more than adequate to drive the controller with fixed data tables: Why not take advantage of it? A very reasonable solution is to dedicate your Scelbi, Mark 8, or what-have-you-kluge to run and refresh the display. The graphics program and refresh vector table would be contained within the 8008 as well as a very simple communications driver which allows it to talk with your new 8080, Z-80, 6800 or what have you. Communication would be between a couple of parallel input and output ports. The 8080 system could contain all the number crunching routines and large programs such as Space War while the 8008 drew and refreshed the playing field or hyper space. Vector table updates would be sent from the main game processor to the 8008 display processor periodically as the game progressed. This system would be fantastic.

Bigger Displays

While 5 inch scopes are quite adequate, full appreciation of vector graphics is realized only on large screen displays. Commercially sold large screen XY displays start at about $600. Hal Chamberlin, in his *Computer Hobbyist* articles mentioned earlier, describes how such a display can be built (with some difficulty) using television style display mechanisms. His work was an invaluable help to me in getting my display going; the circuits you see in this article are my adaptations of some of the circuits Hal published in his excellent publication in connection with his vector display design. *The Computer Hobbyist* is located at Box 295, Cary NC 27511. At last report, back issues were available.

Another possible source of big screen displays is auctions of electronic equipment (often attended by dealers in surplus). For example, *Electronic News* often publishes

Table 2: Refresh Vector Table for Drawing the Star Ship Enterprise. This table contains the information needed (in octal) to generate the Enterprise on a vector display. The addresses are relative to page 001, starting at location 50, for use with the program of listing 2. The end of the display string is indicated by the octal code 377 found stored at location 330. If you were to put some extra imaging information in the picture, the additional vectors would begin at location 330 instead of the end code. The fine print numbers in figure 4 near each line segment are the addresses of the appropriate vector specification in this table.

Vector Address	start X	start Y	end X	end Y	commentary
050	004	104	014	070	Begin Enterprise's bridge
054	014	070	110	070	
060	030	070	054	060	
064	054	060	074	070	
070	110	070	120	104	
074	120	104	004	104	
100	014	104	040	110	
104	040	110	120	104	
110	040	110	042	114	
114	042	114	064	114	
120	064	114	070	116	
124	046	114	050	120	
130	050	120	060	120	
134	060	120	062	114	
140	074	070	104	034	Bridge connecting strut line
144	104	034	064	030	Main body outline
150	064	030	064	010	
154	064	010	104	004	
160	104	004	140	004	
164	140	004	210	014	
170	210	014	204	024	
174	204	024	170	027	
200	154	032	140	034	
204	140	034	104	034	
210	140	034	114	076	Bridge connecting strut line
214	074	032	074	006	Phaser weapons bank
220	064	020	054	030	
224	054	030	054	010	
230	054	010	064	020	
234	054	020	050	020	
240	134	104	134	130	Engine pod outline
244	134	130	320	130	
250	320	130	304	104	
254	304	104	304	130	
260	304	104	134	104	
264	170	104	170	024	Engine pod support strut
270	170	024	154	024	Engine pod support strut
274	154	024	154	104	Engine pod support strut
300	140	104	140	130	
304	134	130	130	124	
310	130	124	126	120	
314	126	120	126	114	
320	126	114	130	110	
324	130	110	134	104	
330	377				End of picture code

REFERENCES

Worth noting are two articles which contain information on vector graphics experimentation beyond that contained in this issue of BYTE:

Concerning the conversion of the Sanders 708 see:

Ciarcia, Steve and Robertson, Carl L: "Simple Graphics Terminal Using Inexpensive Surplus Equipment," *73 Magazine*, September 1976, pages 116-123. (The circuits shown in figure 2 of the present article are revised versions of some of the circuits in author Ciarcia's earlier published work cited here.)

Concerning the generation of character graphics for a vector display see:

Lerseth, Richard J: "A Plot Is Incomplete Without Characters," BYTE, July 1976, pages 64-72.

Concerning experimentation with a magnetic deflection CRT display built from scratch, see:

A series of graphics articles by Hal Chamberlin in Volume 1, issues 1, 2 and 3 of *The Computer Hobbyist.*

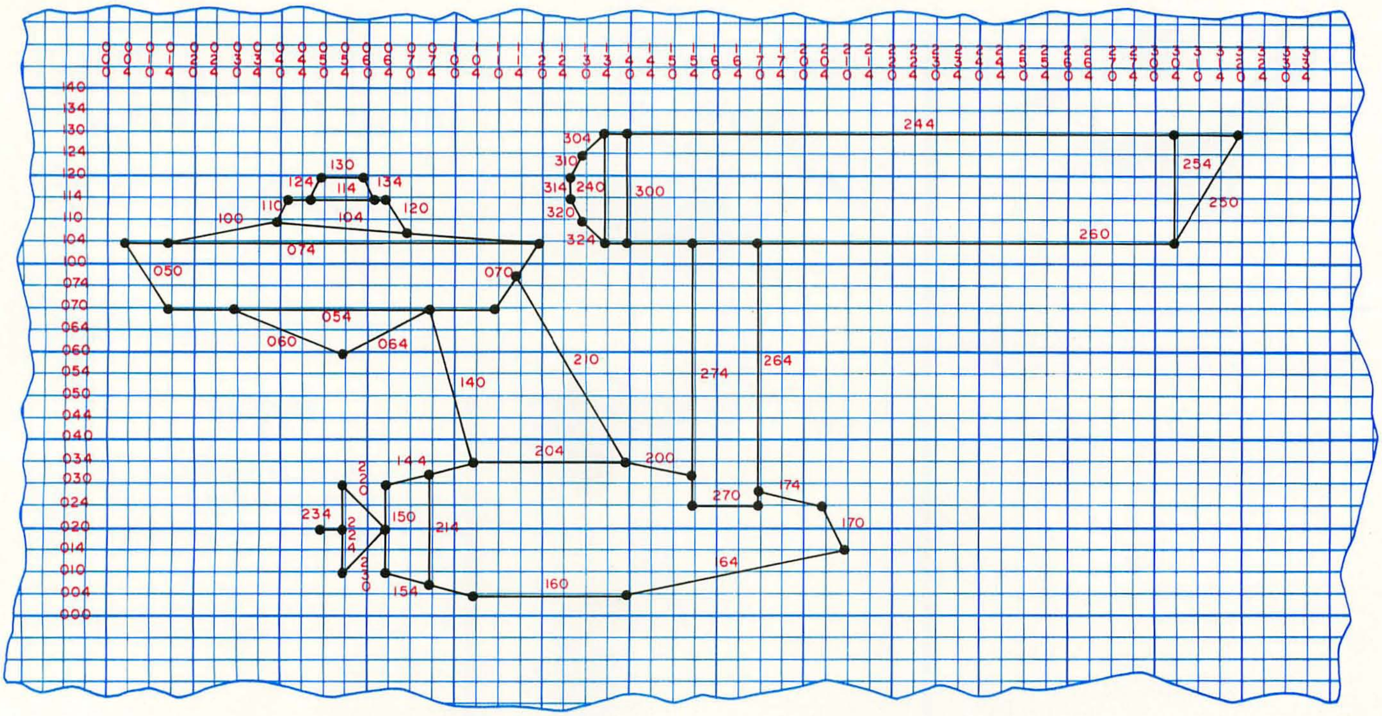


Figure 4: The Star Ship Enterprise, drawn on a graphing grid. One way to prepare data for a display such as this is to plot it out ahead of time on a coordinate grid, such as graph paper. This figure was constructed using the data of table 2; but in the normal procedure the figure would be drawn first, and then one would prepare the table of vector specifications.

auction announcements when they are held; and for those of us in the New England area, the *Boston Globe* auctions section has advertisements of electronic companies' public auctions from time to time.

In my own case, I purchased a surplus Sanders Associates 708 CRT terminal from a local surplus house. This terminal, sold through the 1960s and early 70s, is an all solid state video terminal which was used for alphanumeric display, typically in medical information systems. The terminal used a starburst generator for alphanumerics. This fact alone is enough for one to realize that it is in fact an analog vector scope. The 708 CRT has a 12 inch screen and a P31 high persistence phosphor which makes for some tremendous graphics. I got mine for less than $200, and I found that it was readily converted to a 12 inch XY vector scope for this purpose.

Whatever you choose as a display medium, whether it is a 5 inch oscilloscope or a 12 inch XY display tube, you will find vector graphics both challenging and fascinating. Printing out a message describing a battleship on an alphanumeric CRT is interesting, but it cannot hold a candle to a system which can draw a battleship complete with guns and flag on the screen. Graphics opens a new dimension for the home computer experimenter and it need not be prohibitively expensive. Using high performance peripherals such as this serves to complement a well designed computer system, and provide uses for oscilloscopes when everything is working right with your hardware.■

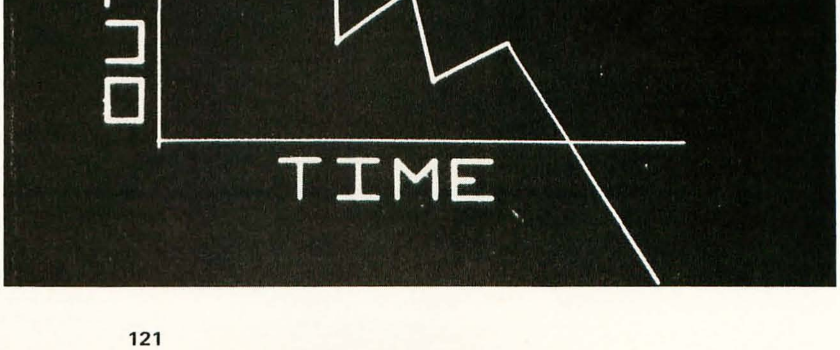


Photo 3: Using the graphics display for engineering outputs. This is a concocted display to illustrate XY plotting situations. Whatever "output" is, it varies with "time" in a ragged sawtooth defined by several vectors.

Continued from page 24

function with a del and attempt to execute it:

```
        NEW                         ←APL
TABLE OF SQUARES                    ←APL
1 1                                 ←APL
TABLE OF SQUARES                    ←APL
2 4                                 ←APL
```

Something is wrong, so you reenter the function definition mode and list your function:

```
       ∇NEW                         USER →
[6]                                 ←APL
       [⎕]                          USER →
       ∇NEW
[1] N←0
[2] 'TABLE OF SQUARES'
[3] N←N+1                           ←APL
[4] N;N*2
[5] →(N≠10)/2
```

The line which used to be 2 is now 3, since APL renumbers all lines to integers when you close the function. You could change the 2 to a 3 in the fifth line; however, there is a better way. If you have to insert or delete lines from your function again, all places where there is a branch arrow may also have to be changed. If APL did not provide a way around this problem, the language would not be as popular as it is. In a function you can specify a variable to become a label by putting it at the beginning of a line and following it by a colon. Whenever you execute that function the labels in it are automatically assigned the value of the lines where they are located. The function when corrected using a label would look like this:

```
       ∇NEW
[1] N←0
[2] 'TABLE OF SQUARES'
[3] LOOPLABEL: N←N+1                ←APL
[4] N;N*2
[5] →(N≠10) /LOOPLABEL
```

The function NEW above is a niladic function, which returns no result. Niladic means it has no arguments. Here is an example of a niladic function which returns a result:

```
       ∇RESULT←THISFUN
[1] X←A+B                           ←APL
[2] RESULT←ρX
       ∇
```

The function could be called inside an expression and used wherever a variable could be used. For example:

```
       A←1                          ←USER
       B←1 2 3 4                    ←USER
       3+THISFUN                    ←USER
7                                   ←APL
```

THISFUN calculates the sum of A and B then finds the length of the sum and returns it as the value of THISFUN. In the function

Proposed Micro APL Specifications

Operators:

All monadic and dyadic scalar operators will be implemented (except for the exponential, logarithmic and circular if space does not permit).

The following monadic mixed operators will be implemented: ravel, index generator, shape. The following will probably be implemented: grade up, grade down, execute, and reversal. Catenation, index of, reshape, and compression will be the dyadic mixed operators implemented, probably along with: take, drop, element of, rotate, and perhaps encode and decode if space permits.

The only composite operator implemented will be reduction.

Data Structures:

Numeric and character data will be allowed, but all numbers will be stored in IBM 360/370 single precision floating point format. Character data will be stored one character every four bytes to conform with floating point format. The space allocated for character and numeric data will be 1 to 2 K, so the maximum number of elements in all variables and internal variables created by evaluation of an expression must be less than 256 to 512. Vector will be the only type of array allowed.

Functions:

Up to eight functions may be defined in workspace. The total memory for function storage will be around 1 to 2 K. Although the functions are stored in essentially the same format in which they are defined, a number of pointers also have to be stored in this area. Recursive functions and local variables may or may not be allowed depending on the amount of memory available.

Total Memory Requirement:

12 K 8008 system. An 8080 or 6800 version could probably run in 8 K by reducing the amount of memory allocated to the workspace.

the value that will be returned is assigned to the dummy variable specified in the function header (in this case, RESULT).

A function could also be monadic, as this function which finds the cosine of a number:

```
     ∇C←COS X
                              } ←— APL
[1] C←2○X

     ∇
```

A monadic function which returns a result, such as COS, could be used anywhere a monadic operator is legal. (See table 1 for the meaning of symbols used.) Thus APL allows the programmer to effectively create new APL operators, which sometimes adds the readability of a program. An example of the use of this particular function is:

```
     (COS ○2) + (COS ○3)
```

This is certainly more readable than

```
     2○○2 + 3○○3
```

which would calculate cos (2π) + cos (3π). A dyadic function which returns a result could be used anywhere a dyadic operator is legal, so the function ADD could replace +:

```
     ∇AB←A ADD B
                              } ←— APL
[1] AB←A+B

     ∇

     34 ¯2 ADD 4 6            USER —→

38 4                          ←— APL
```

Table 2: Typical APL System Commands. These are the global instructions used to control the interpreter program. A small system APL would use versions of these commands.

)CLEAR	Clears all functions, variables, etc.
)LOAD	Loads a new workspace from an auxiliary storage device.
)SAVE	Saves a workspace on an auxiliary storage device.
)FNS	Lists all functions in the current workspace.
)VARS	Lists all variables in the current workspace.
)SI	Displays the state indicator (internal information about where execution stopped, and where errors have occurred).
)SI CLEAR	Clears the state indicator.

A dyadic or monadic function need not return a result such as this function:

```
     ∇AA SUM BB               ←— APL

[1] 'THE SUM IS';AA+BB        ←— APL

     ∇                        ←— APL

     3 SUM 4                  USER —→

THE SUM IS 7                  ←— APL
```

However, it is illegal to use a function which does not return a result inside an expression.

The previous discussion has shown what the typical APL functions are like. In addition, there are usually several system commands associated with an APL interpreter. Table 2 gives a short listing of several typical system commands. These refer to the means of loading, clearing and saving the APL workspace, catalogs of what is in the workspace, etc.

This should give you a good idea of the power of APL on a large system, but how useful would APL be on your small system byte banger? Obviously many of the operators would take a good deal of the memory of an average small system to implement (matrix divide, for example). Another problem is that since APL is so generalized in its operations, it is somewhat memory inefficient. For example, to add two vectors, A and B, together, all you have to type is:

```
     A+B                      USER —→
```

but, in order to calculate the sum, APL has to make a copy of each of the vectors first; so if the length of A and B is 64 (256 bytes at 4 bytes per floating point number), then 1 K would be in use during this operation. Using a comparable program in BASIC would require 516 bytes, since the sum could be calculated one element at a time, printing each sum out as it was calculated. There is also an IO problem, since most people have ASCII encoded terminals (or the equivalent), terminals which do not have the entire APL character set on them.

The only way I see that APL could be practical for a memory and IO limited personal processor is if the features were limited (only vectors, for instance, instead of up to 63 dimensions for arrays), and if the number of operators were kept to a workable minimum. All the scalar operators are performed in a similar manner, so they could be easily implemented. But only the most powerful of the mixed operators should be implemented, for example:

```
     ι   ρ   /   ⌽   ↑   ↓

     ⍋   ⍒   ε   ?   ,
```

Of the composite operations, reduction is the most used, and easiest to implement. Whether or not character data would be allowed depends on what size system was available. An easy way to handle character data would be to store one character every four bytes, so the routines used for numbers would not have to take care of the special case of character data. Although this method would waste memory in the workspace proper, it would greatly reduce the size of the interpreter, and in a small system this is where most of the memory is used. On large systems, logical (1 and 0) data is stored as bit level data for memory efficiency, but on a small system the major programming problem of unpacking bits would probably not be worth the small savings it would provide.

GLOSSARY

Array: An array is a group of numbers (or characters) which in APL can be subscripted or treated as a single object. Vectors are the only type of arrays discussed in this article; however, a full implementation of APL allows matrices and up to 63 dimensions for arrays.

Character: As an APL data type, character is the kind of operand which is allowed for most mixed operators, but not for scalar operators other than = and ≠.

Command: An APL system command causes some operation to occur, such as destroying the current workspace, or loading a new workspace from an external device. A command begins with a right parenthesis and has an English keyword, such as) CLEAR or) SAVE. A command is normally legal in either execute or function definition mode.

Compiler: A program which translates a high level source program into machine language object code is called a compiler.

Composite operator: A generalized operator which takes a built-in operator(s) and performs it (them) in a special way. An example is reduction, which takes a scalar operator and performs it upon all the elements of the operand, right to left.

Dyadic: A mathematical term meaning having two operands. An APL function or operator which is dyadic must have an operand on both sides of it.

Execute mode: This is the APL system mode where an expression such as 1+1 will print the result. The del is used to leave the execute mode for the function definition mode.

Function: This is the name of an APL program. Since a workspace may have more than one function and a function can be niladic, monadic, or dyadic, and may or may not return a result, functions can be used as main programs, subroutines, or functions.

Function definition mode: This is the mode in which APL functions are created. The del is used to go back and forth between the execute and function definition modes.

Function header: This is the first line of a function, before the actual program starts. It gives information to the interpreter about the function: Does the function return a result? What, if any, local variables are used in the function?

Interpreter: APL is implemented as an interpreter, a program which takes a source listing of the high level program and performs operations as the meaning of the program is being translated.

Mixed operator: In APL a mixed operator is one that allows or requires the operand(s) not to obey the rule for the size and shape that scalar operators do require.

Monadic: A function or operator which has only one operand to the right of the operator.

Niladic: A function which has no operands.

Operator: An operator is a built-in function which performs some common task such as addition, and is represented by a single, reserved, character.

Quad: The quad is used in expressions to perform input and output, and in the function definition mode to list the function. To the right of an assignment arrow quad will request an expression, which will be evaluated, and the result used in place of the quad in the expression. To the left of an assignment arrow, quad prints out the value assigned to it. Quote-Quad is for input of character string data only.

Rank: This is the shape of the shape of an array; in other words the number of dimensions of the array.

Scalar: A single value, such as 8 or ¯23.4. Scalars can be used with dyadic scalar operators to operate on all the elements of the other operand.

Scalar operator: An operator which obeys this rule: If the size of the two operands (for the dyadic operation) is the same, do the operation on the values in parallel. If one or the other is scalar, do the operation between the scalar and all the elements of the other operand.

Shape: In a full implementation of APL, this is a vector which represents the dimension of an array.

Subscript: This points to which elements of an array are to be used. APL allows subscripts to be vectors, a feature most other languages do not allow.

Variable: A symbolic name given to a value, such as 3, or ¯4. A local variable (dummy) is one which will be destroyed after the function in which it is in finishes executing.

Vector: A one dimensional array, the only type of array which can be represented as a constant in APL.

Workspace: The APL name for all the functions, variables, and internal state indication data which the user has created.

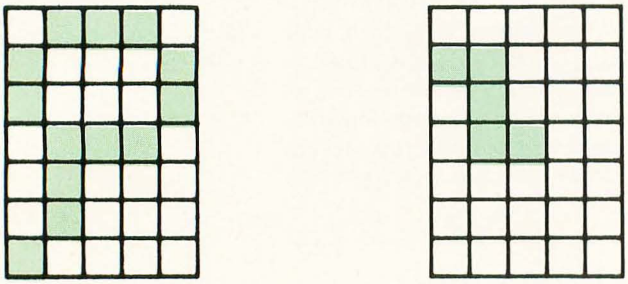


Figure 1: Two examples of alternate dot matrices for APL characters. The character at the left is the Greek character rho, shown in an approximation which distinguishes it from a Roman P. The character at the right is a Greek iota.

The storage problem of A+B could easily be avoided by writing a program similar to the one in BASIC:

```
      ∇NOPROBLEM
[1] N←0
[2] LOOP: N←N+1
[3] A[N]+B[N]
[4] →(N≠ρN)/LOOP
      ∇
```

← APL

In APL the programmer has the advantage of being able to choose whether or not he or she wants to use the built-in vector operator or program it, which is something that the programmer of most small system BASICs does not have.

As far as the character set problem is concerned, a translation table could be built to translate ASCII input like $R (control R) into the internal code for ρ, or $I into the internal code for ι. Once translated, the computer would not care if the ι came from an APL terminal or even a Baudot Teletype. Or, an ASCII keyboard could be used when the internal codes for the APL character set were chosen so that the positions of the characters on the ASCII keyboard simulated where the same key was on the APL keyboard. An upper case I would be ι while i would be translated into I. The same translation would occur with R and ρ; r and R. On such a system a TV display could be converted to show a 5 by 7 representation of the character, as shown in figure 1.

Of course, software would have to handle the translation from the internal codes to the graphics codes, and software would have to take care of the cursor; so that if the APL programmer backspaced to create a legal overstrike, such as ' over . to form !, or ○ over | to form ⌽, the rotate operator, it would appear as if the overstrike actually occurred. (On most CRTs, it would not have; but rather it would have been replaced by a new code, created by the software as soon as the backspace and corresponding overstrike were entered.)

Some people love APL and some people hate it. I think it is one of the best interpreters around, next to BASIC. I hope to be able to implement a subset of APL on my computer in my spare time so that it will speak more than one language. I'm not sure how powerful a small system APL would be, but I will report my success or failure in a later article. In the meantime, I'd like to hear from anybody who is interested in or knowledgeable about small systems APL.■

BIBLIOGRAPHY

John Coake and J T Schwartz, *Programming Languages and Their Compilers*, Courand Institute of Mathematical Science, New York, 1970, pages 601-642. This book describes the methods of writing a compiler for various languages, including an involved chapter on APL.

Leonard Gillman and Allen J Rose, *APL 360: An Interactive Approach*, Wiley and Sons, New York, 1974.

Kenneth Iverson, *A Programming Language*, Queens University, Kingston, Ontario, 1968. This is a fairly mathematical type book, and does not represent APL as it was implemented, although Iverson is the creator of APL.

Harry Katzan, *APL Programming and Computer Techniques*, Van Nostrand Reinhold, New York, 1970.

Harry Katzan, *APL User Guide*, Van Nostrand Reinhold, New York, 1971.

Sandra Pakin, *APL/360*, Science Research Associates, Chicago, 1972. This is a manual for the IBM 360 implementation of APL.

Sandra Pakin, *APL: A Short Course*, Prentice Hall, Englewood Cliffs NJ, 1973.

William Prager, *Introduction to APL*, Allyn and Bacon, Boston, 1971.

Xerox APL Reference Manual, Xerox Corp, El Segundo CA 1973. This is where I got the idea of mnemonic representation of APL characters using the $.

BYTE'S BITS

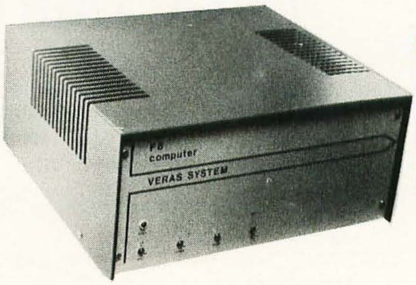
Hot Off the Grapevine

Texas Instruments has announced (in marketing literature of somewhat limited circulation) a new single chip video controller integrated circuit which is envisioned as part of the 9900 family line. This is a single chip generator for all the timing and control information needed to run a raster scan display interfaced to an LSI processor's bus. The grapevine says that it is being sampled at the present time, and will be on distributor shelves next spring. The same grapevine identifies a second, similar controller chip announced by another company as well. Either item would make excellent material for a tutorial review article in BYTE since these chips have obvious uses in personal computing systems.■

Editor's Pet Peeve: All you wonderful users of the English language, engineering subset, take note: A "baud rate" is a redundant abomination. The concept is *data rate*, sometimes measured in *units of* baud which we take to signify bits per second. (The original definition of a rate *measured in baud* is a little more complex.)■

WHO DARED TO CHALLENGE THE 8080 & 6800 SYSTEMS?

WE DID!



7" x 16" x 14-3/4"

VERAS SYSTEMS "F-8"

A COMPLETE SYSTEM FOR ONLY $459.00 KIT FORM

$709.00 ASSEMBLED and TESTED

COMPARE STANDARD FEATURES ON BASIC SYSTEMS
YOU'LL SEE WHY THE VERAS F-8 SYSTEM WINS HANDS DOWN!

*Registered Trademarks of Fairchild and Motorola

STANDARD FEATURES	VERAS F8	8080 & 8080A SYSTEMS	6800 SYSTEM
Parallel I/O Ports	(3) 8 Bit Bidirectional Brought Out To Rear Panel Conn's.	None	None
Serial I/O Port	RS-232 or TTY Brought Out To Rear Panel Term. Strip.	None	RS-232 or TTY
Interval Timers	(2) Programmable Interval Timers	None	None
Interrupts	a) Vectored Interrupt To Location 0090 Hex. b) Vectored Interrupt Programmable Location c) Two Vectored Interrupts Associated With Interval Timers d) Total of (4) Interrupts In A User Defined Priority Interrupt Structure	None	a) 2 Non Vectored Interrupts on P I A b) 2 Vectored S W I & N M I c) Total of 4 Non Prioritized Interrupts
Built In Mini Operating System in ROM For Terminal And Memory Debug	FAIRBUG*	None	MIKBUG*
Loader Program	Automatic Internal ROM	Manual Console Switches	Automatic Internal ROM
Static RAM Memory	1024 BYTES	None	2048 BYTES
Card Rack	Rugged Alum. Self Contained Card Rack/Plastic Self Aligning Card Guides	Card Supports	None
Auxiliary DC Power To Power Peripherals	+5V, -5V, +12V, -12V @ 1 Amp. Ea. Regulated At Rear Panel Terminal Strip	None	None
Basic Kit Price	$459.00	$539.00, $599.00 or $840.00 Depending On System	$395.00

OUR 4K STATIC RAM BOARD FEATURES: (OPTIONAL)

- On board decoding for any four of 64 pages.
- Address and data lines are fully buffered.
- No onboard regulators to cause heat problems. (Chassis mounted)
- 4K memory boards with connector, buffers, static RAM's & sockets are available in kit form **$149.00**

The More Flexible and Expandable Computer at a Comparative Price.

VERAS SYSTEMS

Warranty: 90 days on parts and labor for assembled units. 90 days on parts for kits. Prices, specifications, and delivery subject to change without notice.

THE VERAS SYSTEM

The VERAS System can be made into a 17K processor by merely adding four of our optional memory boards. The kit includes everything you need to build the VERAS F-8 Computer as described. All boards, connectors, switches, discrete components, power supply and cabinet are supplied. Programming manual, data book and simplified support documentation supplied, 8K Assembler and Editor (paper tape or K.C. std. cassette) available on request with minimum order of 8K RAM.

THE CPU BOARD FEATURES:

- Two I/O ports each on the CPU and ROM chip make 32 bidirectional TTI lines.
- The Fairbug* programmed storage unit provides the programmer with all I/O subroutines, allows the programmer to alter or display memory, and register its contents via teletype.
- Programmable internal timer is built into the ROM chip.
- Built in clock generator and power on reset are built into the CPU chip.
- There is a local interrupt with automatic address vector.
- It is expandable to 65K bytes of memory.
- 20 mil loop and/or RS232 interface included.
- 1K of on board 2102 RAM.
- Serial interface built into PSU chip.

*Fairbug is a registered trademark of Fairchild Corp.

TINY 2 K BASIC (AVAILABLE) OCT. 15, 1976 $25.00
FULL BASIC (AVAILABLE) DEC. 15, 1976 $50.00

Computer dealers and hobbyist club inquiries are invited. Expected delivery time 30 days or less.

VERAS SYSTEMS
A Div. of Solid State Sales, Inc.
Box 74B, Somerville, MA 02143
(617) 547-1461

☐ Enclosed is check for $ ______

or ☐ Master Charge # ______

☐ VERAS F-8 Computer Kit ☐ Assembled

☐ 4K RAM Board Quantity ______

Name ______

Address ______

City, State ______ Zip ______

It's all right here

- Prompt service
- In-depth inventories

BRIGAR electronics

ELECTRONICS for INDUSTRY

LOW, LOW PRICES!

PHONE ORDERS
607 723 3111

Offices & Warehouse: 10 Alice Street, Binghamton, New York 13904 • Tel.: 607-723-3111

SAVE SAVE

MINIATURE LAMPS



INCANDESCENT LAMPS

CHICAGO MINIATURE LAMPS #387
SAME AS #327 EXCEPT LONGER LIFE

28 volts. .04 amps. Bulb Style T-1-3/4 Mid-Flange Base. BRAND NEW BOXED, 100 per box. 40,000 pcs in stock.

PRICE SCHEDULE
1 - 99 .. 40¢ each 100 - up .. 35¢ ea.

NEON INDICATING GLOW LAMPS



NE-2 neon lamps, with dropping resistor to operate directly from 115 VAC. Nominal Watts - 1/17. T-2 Clear Bulb, wire terminals.
25,000 pcs in stock. Mfd by G.E. Co.
10¢ each up to 1000 pcs.
1000 pcs & up 7¢ each.

INDICATOR LIGHTS

BRITE GLO INCANDESCENT LAMPS



BRITE-GLO Incandescent Lamps
Ultraminiaturized Series, 100,000 hr.life Versatile, low cost indicator lamp. Ideal for Stero Amplifiers, Mobile Equipment, Instrumentation, Computer and Display Panels, Etc.
High temperature, T 1-3/4 lamps - Metal Base - Wire Terminals. Metal base increases light output and protects lamp from damage.
41, 685 pcs - 6.3 volts, 75 ma, 8" wire leads.
40,263 pcs - 6.3 volts, 200 ma, 13" wire leads.

DISCOUNT SCHEDULE

SALE	1 - 99	25¢ ea.	SALE
	100-499	20¢ ea.	
	500-999	15¢ ea.	
	1000-UP	12¢ ea.	

BARGAINS PRECISION RESISTORS



Type RS. These should be specified when space is at a premium and when precision and power are needed. They are of the conventional type with pigtail mounting leads. They are completely protected and silicone sealed. Surpasses MIL R-26

BRAND NEW — FINEST QUALITY

QUAN	MFR	TOL. (%)	OHMS	WATTS	PRICE Each
10,686	T.I.	1	191	1/8	$.04
3,000	Electra	1	191	1/8	.04
29,000	I R C	1	243	1/8	.04
75,000	T.I.	1	2.43K	1/8	.04
35,000	T.I.	1	909	1/4	.06
7000	T.I.	1	10	1/2	.07
3000	Electra	1	15.4	1/2	.07
5700	Electra	1	28	1/2	.07
7900	I R C	1	53.6	1/2	.07
7200	I R C	1	90.9	1/8	.04
6600	I R C	1	1650	1/8	.04
1800	I R C	1	6190	1/8	.04
1000	I R C	1	2670	1/8	.04

Computer Grade Capacitors



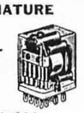
MALLORY — PRICE SLASHED

CG (STANDARD) 85°C Supplied with PVC Insulating Sleeve

Quan.	Mfd.	Vdc	Size	Each
10000	1100	35	1-25/64 x 2-11/64	$.75
25000	100	350	1-25/64 x 2-11/64	.50
10000	80	400	1-25/64 x 2-11/64	.60

POTTER & BRUMFIELD RELAYS

SUB-MINIATURE GENERAL-PURPOSE



TYPE KH 5505 (Identical to KHU17D11)
COIL: 24vdc, 650 ohms.
CONTACTS: 4PDT, 3amps at 30vdc or 120 vac

Small rugged DC relay with clear polycarbonate dust cover. Contacts are gold flashed silver.
Mounts by solder terminals or 3-48 stud 5/16" long. Also mounts as plug-in.
6500 pc in stock, BRAND NEW, while they last at this price. $1.25 each

LAMINATED PHENOLIC PLASTIC PANELS



BRAND NEW PHENOLIC SHEETS
Grade LE- Natural
Type - FBE, Mil Spec: 15035, .062" thick Needed by all experimenters, amateurs and radio men for insulating parts - terminal strips - building sets or for research & development, etc.

Stock # 560 - 6" x 6" x .062" 42¢ each
" # 561 - 6" x 12" x .062 82¢ each
" # 563 - 12" x 12" x .062 $1.58 ea
" # 564 - Full sheets 38" x 47" x .062" $13.00 each.

BRAND NEW QUALITY SHEETS

High-Speed Reset Counter Veeder-Root

4 FIGURE COUNTER MANUAL RESET
115VDC/115VAC
VEEDER ROOT 177604



115VAC (Rectifier supplied). Basic counter is 115VDC, 6 watts 4 figure, black on white background. Push button reset. Dim.: 1-7/8" width 1-1/8" high x 2-1/2" deep.

$12.95

LIGHTED PUSHBUTTON SWITCHES

$2.95 Each



Unit can be used on 6 v or in series for 12 volt SPST, mom, PC board gold plated pins for wiring or easy installation.
10,000 pcs in stock

Mfr: Grayhill Switch Co.
2" x 1½" x 1½"

Widely used on computers, machine tools, office machines, guidance control and a wide variety of applications.

Electrolytic Capacitors

TUBULAR, HIGH RELIABILITY



QUAN.	MFD	VDC	EACH
600	100	3	.30
375	250	3	.35
298	500	6	$1.00
299	1000	6	1.00
571	1200	6	1.02
375	1500	6	1.10
313	2000	6	1.11
597	100	12	.60
4000	50	15	.30
26,000	200	15	.30
416	1500	15	1.10
350	2000	15	1.10
2500	10	25	.25
1161	2000	25	1.40
2600	10	50	.40
1000	500	50	.80
825	100	50	.95
1099	150	50	.90
6549	250	50	.98
1942	300	50	1.07
10,500	60	150	.70
520	80	150	.90
290	300	150	1.00
476	40	200	1.00
409	100	250	1.39
284	200	250	1.50
255	5	350	.90
281	10	350	.90
20,000	4	250	.35
339	8	250	.70
400	16	250	.60

QUAN.	MFD.	VDC	EACH
1055	30	350	$1.00
246	40	350	1.14
967	4	450	.98
400	10	450	1.07
529	15	450	1.07
453	20	450	1.14

METALLIZED PAPER FILM

QUAN	MFD	VDC	EACH
899	.1	100	$ .35
437	.33	100	.40
400	1.0	100	.45
1360	.22	150	.40
820	.022	200	.35
4200	.033	200	.35
400	.1	200	.40
771	1.0	200	.40
417	1.5	200	.40
1100	.001	400	.35
1100	.0022	400	.40
1600	.022	400	.45
565	.047	400	.50
4000	.022	600	.25
389	.05	600	.30
500	.1	600	.30
600	.2	600	.30
717	.25	600	.30
30,000	.01	100	.15
1000	.047	100	.20
291	.22	400	.30

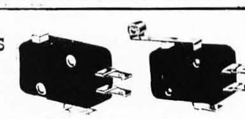


NEW!

HIGH POWER DUMMY LOAD

Mfg. Eastern Microwave. #10-1254...RF Coaxial Load Resistor utilizing water flow to cool and provide a load resistance of 50 ohms for inputs from DC to 4000 MHz. Maximum power input is 1000 watts. Requires a flow of .218 GPM min. per 100 watts of input power; 100 psi pressure. Compression fittings on water line inlet and outlet for copper tubing supply lines. Has a Flange type coaxial fitting, friction type, 13/16" I.D. with 2-1/4" flange. Completely shielded unit of machined aluminum. Size: 1-5/8" diameter by 5-3/8" long. Shipping weight, 2 lbs.
5,000 pcs in stock, NEW BOXED........$14.95 each......

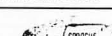
Cherry Electrical Products



These switches are postage stamp sized precision switches which feature long operating life & high electrical capacity for their size 5 amp, 1/4 HP, 125-250 vac, SPDT, quick dis-connect terminals. 2,000 pcs in stock without lever..75¢ each
2,400 pcs in stock with extended roller lever. 95¢ ea

TANTALUM CAPACITORS

SPRAGUE



TYPE 109D

TUBULAR SINTERED-ANODE TANTALEX CAPACITOR
#109D147X9060T2
1620 pcs - 140 uf ±10% - 60vdc $2.00 ea.

GENERAL ELECTRIC — PLAIN FOIL TANTALUMS +125° C (Use CL31 instead of CL30; or, CL33 instead of CL32 for Mylar sleeve) CL30/31—POLARIZED

3,000 pcs CL31BE180MPE, 18uf - 15vdc 65¢ each
300 pcs CL31BE550MPE, 55uf - 15vdc 85¢ each
non-polarized CL32/33
900 pcs CL33BL020MNE, 2.0uf - 75vdc $1.00 each

MALLORY — MTP/MTPH ULTRA-MINIATURE TANTALUM CAPACITORS
MTP TYPES

10,550 pcs MTP685M020P1D, 6.8uf - 20vdc 55¢ each
2,100 pcs MTP106M030P1A, 10uf - 30vdc 75¢ each
1,028 pcs MTP336M010P1A, 33uf - 10vdc 65¢ each

MALLORY — TAS SOLID ELECTROLYTE TANTALUM CAPACITORS

4996 pcs, 4.7uf- ±10%, 6vdc, TAS475K006P1A 35¢ ea
394 pcs, .47uf ± 10%, 6vdc, TAS474K006P1A 38¢ ea
1430 pcs, 47uf ± 10%, 6vdc, TAS476K006P1C 40¢ ea
326 pcs, 220uf ± 20%, 6vdc, TAS227M006P1G $1.20 ea
285 pcs, 5.6uf ± 10%, 10vdc, TAS565K010P1C 40¢ ea
4574 pcs, 10uf ±10%, 10vdc, TAS106K010P1C 40¢ ea
900 pcs, 15uf ±20%, 10vdc, TAS156M010P1C 40¢ ea
887 pcs, .068uf ±10%, 20vdc, TAS683K020P1A 51¢ ea
152 pcs, 2.2uf ± 20%, 20vdc, Sprague 40¢ ea
1457 pcs, .47uf ±10%, 35vdc, TAS474K035P1A 51¢ ea

GENERAL ELECTRIC — PLAIN FOIL TANTALUMS +125° C

365 pcs G.E. 40uf,±20%, 50vdc, 125°C
Type: 2K105AA6M. $1.75 each
1168 pcs SPRAGUE, 40uf, ±20%, 50vdc, 125°C
Type: 220D005A3M. $1.75 each C

All Orders F.O.B. Binghamton, NY.
Minimum Order $5.00
NY State Residents please add sales tax

BRIGAR ELECTRONICS publishes a monthly flyer If you want to be placed on our active mail list, send us your Name ______ Address ______

Computer Grade Capacitors

PHONE ORDERS
607 723 3111

Instant off-the-shelf delivery.



SAVE

QUAN.	MFD.	VOLTS	SIZE	PRICE EA
350	30000	5	1-3/8 x 4-3/16	$1.50
494	60000	5	3" x 4-1/2"	1.50
1016	25000	6	2-1/16" x 4-9/16"	1.50
527	40000	7	2" x 5"	1.50
353	15000	10	2" x 4-1/8"	1.25
869	15500	10	2" x 4-7/16"	1.25
277	40000	10	3" x 4-1/2"	1.70
316	74000	10	3" x 4-3/4"	2.00
375	15000	12	2" x 4-1/2"	1.25
267	14000	13	2" x 4-3/8"	1.25
600	10000	15	2-1/16" x 4-1/2"	1.25
529	12500	16	2" x 4-1/8"	1.25

QUAN.	MFD.	VOLTS	SIZE	PRICE EA
252	11500	18	2" x 4-1/2"	$1.25
628	11000	19	2" x 4-1/2"	1.25
1014	8200	25	1-3/8" x 3-1/8"	1.70
10706	1100	35	1-25/64" x 2-11/64"	1.00
4200	750	60	1" x 3-1/2"	.75
792	1500	80	2" x 4-1/2"	1.30
478	450	100	1-7/16" x 2-9/16"	1.00
337	400	100	1-7/16" x 2-1/2"	1.00
200	500	150	2-1/16" x 2-3/8"	1.00
500	750	200	1-1/2" x 4"	1.00
23436	100	350	1-25/64" x 2-11/64"	.75
11800	80	400	1-25/64" x 2-11/64"	.75

Production Description:



Photo 1.

The Astral 2000

Editor's Note

This descriptive summary is based upon materials supplied by M&R Enterprises, as well as first hand inspection of prototype hardware during a trip to California in April of this year. The marketing and development of the Astral 2000 is, to say the least, cautious and painstaking. It has been essentially kept under wraps until the BASIC facility was sufficiently developed to allow release. The system has been shown off in bits and pieces throughout the design process by people from M&R who attend the sessions of the Homebrew Computer Club in Palo Alto CA. Now that M&R Enterprises is finally to the point of delivering the systems in initial production quantities it leaves the rumor stage and becomes a full fledged contender in the marketplace.... CH

The Astral 2000 system is based upon the Motorola 6800 central processor design. The use of any one of the many microprocessors is subject to numerous design choices in the details of how the chip will be buffered and interact with the rest of the computer world. The designer of the Astral 2000 computer chose to take a route which seems inherent in the design simplicity of the second generation microprocessors such as the 6800. The Astral 2000 bus is a minimum complexity, minimum pinout design which in essence simply buffers the processor pins of the 6800. As a result, it fits into the standard 44 contact (dual 22 pin) edge connectors which are widely available at moderate cost. (This choice of a minimum pinout backplane configuration also makes it possible for this computer to use standard Vector Electronic Co cards as the basis for homebrew peripheral interfaces.) This simplicity of the backplane bus and use of standard connectors is one of the most elegant features of the Astral 2000.

Without indulging in a major review, here in summary is what the 6800 processor chosen for the Astral 2000 will give the programmer in machine assembly language:

- 1 index register (16 bits),
- 1 stack pointer register (16 bits),
- 2 accumulators (8 bits),
- 1 flag register (6 bits active),
- 5 addressing modes for data operations (direct 8 bit, extended 16 bit, immediate, indexed with 8 bit offset, and inherent operands),
- 4 addressing modes for processor control operations (relative with 8 bit signed offset for branches, jumps and subroutine calls, with extended 16 bit addresses or addresses indexed by 8 bit offset, and inherent operands for miscellaneous processor control instructions).

Data control manipulations include the usual and expected operations of a good general purpose instruction set in the second generation of microprocessors. Also worth noting are a decimal adjust instruction to handle packed decimal data and push and pull operations for the accumulators utilizing the stack as temporary data storage. The processor has a software interrupt, a reset sequence, a maskable interrupt line, and a nonmaskable interrupt line; all of these except the reset completely save the processor state prior to the beginning of the new instruction sequence. These interrupt-like operations are vectored to a new program location via 4 pointers located in the last 8 bytes of address space (locations FFF8 to FFFF.) For more complete description of the 6800 architecture see the Motorola 6800 literature.

As implemented in an Astral 2000 system, the first 56 K bytes of a total 64 K bytes in the 6800 memory address space are left open for ordinary programmable memory, the single most valuable resource of any general purpose processor's central processing unit. What this means is that the user can insert cards in increments of 8 K to the point where some level less than or equal to 56 K has been acquired. (This assumes that there are enough card slots in the backplane to accommodate such an expansion.) M&R Enterprises presently has available both an 8 K programmable memory board and an 8 K EROM board (using the 5204 UV erasable chips) with which to take advantage of memory address space. The standard package of the Astral 2000 includes one of the 8 K programmable memory boards. The upper 8 K of a Motorola 6800's address space is reserved for the system monitor and peripherals of the Astral 2000 design.

The Monitor and IO

The basic concept of the Astral 2000 is that of a "turn key" system, even when the unit is purchased in kit form. An aspect of this is the preassembled and tested processor board and memory board. Another aspect of this is in the physical configuration seen in the photographs. The front panel may seem at first sight to be an unimaginative repeat of the traditional front panel seen in computers from time immemorial. But it is not essential, and is put in the design apparently for

FAIRCHILD ANNOUNCES THE **FAIRCHILD**

SOLID STATE TECHNOLOGY KIT

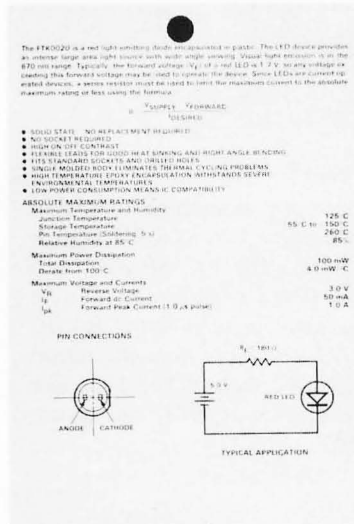
- FOR THE EXPERIMENTER WITH TASTE FOR "STATE OF THE ART" PRODUCTS
- COMPLETE SPECIFICATIONS ARE PRINTED ON THE BACK OF EACH TECHNOLOGY KIT

FTK0020



CARD FRONT

FTK0020



CARD BACK

FTK0106



CARD FRONT

DIGITS

FTK0001	0.5" High Common Cathode Digit	$1.00
FTK0002	0.5" High Common Anode Digit	1.00
FTK0003	.357" High Common Cathode Digit	.75
FTK0004	0.8" High Common Cathode Digit	2.00
FTK0005	0.8" High Common Anode Digit	2.00

0.8" HIGH DISPLAY ARRAYS

FTK0010	12 Hour, 3½ Digit Clock Display	7.00
FTK0011	24 Hour, 4 Digit Clock Display	8.00

LED LAMPS

FTK0020	10 Red LED Lamps	1.00
FTK0021	5 Mixed Colored LED Lamps	1.00
FTK0022	10 LED Mounting Clips	1.00
FTK0023	5 Three Piece LED Mounting Adapters	1.00

PHOTO TRANSISTORS

FTK0030	5 Flat Lens Photo Transistors	1.00
FTK0031	5 Round Lens Photo Transistors	1.00
FTK0032	3 Flat Lens Photo Darlingtons	1.00
FTK0033	3 Round Lens Photo Darlingtons	1.00

PHOTO ARRAYS

FTK0040	9-Element Tape Reader Array	16.00
FTK0041	12-Element Card Reader Array	24.00
FTK0042	Reflective Opto Coupler	4.00

COUPLERS

FTK0050	3 General Purpose Opto Couplers	1.00
FTK0051	Darlington Opto Coupler	1.00

MOS CLOCK CIRCUITS

FTK0400	Digital Clock/Calendar Circuit (FCM7001)	7.00
FTK0401	Digital Clock/Calendar with BCD Outputs (FCM7002)	7.00
FTK0402	Direct Drive Digital Clock Circuit with AC Output (FCM3817A)	5.00
FTK0403	Direct Drive Digital Clock Circuit with DC Output (FCM3817D)	5.00
FTK0405	Direct Drive Digital Clock/Calendar Circuit (FCM7015)	6.00

KITS

FTK0106	Automobile Clock Kit	40.00

- THESE PRODUCTS ARE PACKAGED FOR OUTSTANDING WALL DISPLAY APPEARANCE
- FULL FAIRCHILD PRODUCT LINE TO FOLLOW
- DEALER'S AND WHOLESALER'S INQUIRIES INVITED—PRICE LIST AVAILABLE.
- BUY WITH PRIDE THE PRODUCTS BUILT BY THE INDUSTRY'S LEADER — FAIRCHILD

Satisfaction Guaranteed. $5.00 Min. Order. U.S. Funds.
California Residents — Add 6% Sales Tax
Send a 24c Stamp (postage) for a FREE 1977 Catalog.

JAMES

NOW OPEN SATURDAYS

1021 HOWARD STREET
SAN CARLOS, CA 94070
PHONE ORDERS — (415) 592-8097

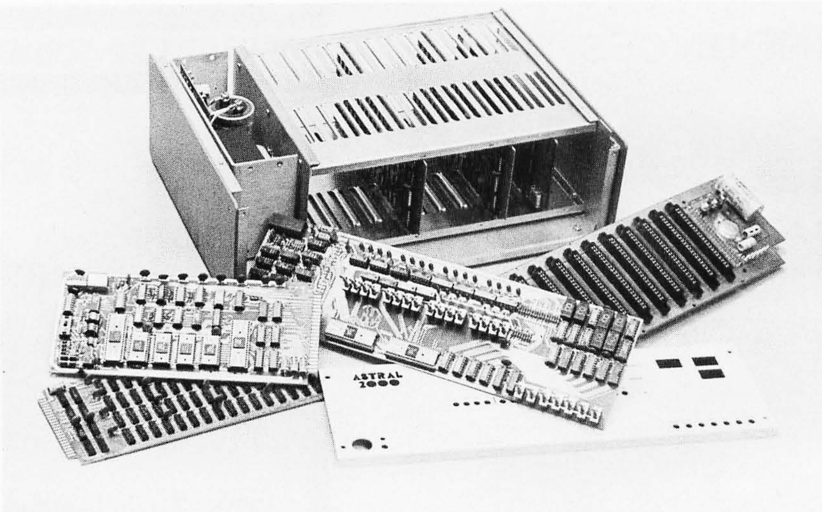


Photo 2: The Astral 2000 computer is $995 partially assembled or $1250 fully assembled and tested. Additional fully assembled 8 K RAM boards are $245 each and the 8 K EROM board is $59.95. The VID-80 video terminal is $189.95 in kit form or $245 fully assembled and tested. IO tape interface units are $49.95 each. The software package includes Astral BASIC on magnetic tape cassette or paper tape (specify), complete documentation and a free one year subscription to the Astral Newsletter, *all for $35. The Astral 2000 is available from M&R Enterprises, POB 61011, Sunnyvale CA 94088.*

purposes of imagery and showing off a classical "computer." The toggle switches and lamps don't really control a thing directly, since they lead directly to a group of PIA ports, as do the other "front panel" features such as the system's real time clock.

The "turn key" effect is achieved by two subsystems of the central processor board design and an optional VID-80 video display terminal board. The processor board contains the key to turn key operation, a 2 K byte ROM monitor. This monitor program can be used in one of two modes: It can drive the VID-80 terminal interface board, or a standard serial line interface with both 20 mA current loop (Teletype compatible) and EIA RS-232C compatible drivers for typical commercial terminals. The serial terminal interface also includes control lines for auxiliary functions such as reader and punch control.

The logical features of the Astral 2000's monitor program will prove quite useful. The input of the monitor is buffered so that commands can be repeated simply by entering an ASCII space code from the keyboard. In the context of a command like the one which dumps "n" bytes, this can be a saver of keystrokes: once "TYPE 8" has been entered, additional 8 byte groups of data can be dumped in sequence by simply entering a space. The monitor also includes machine language debugging facilities to aid the low level language hacker. These debugging facilities include patching and displaying memory contents, insertion of break points, stepping through break points, etc. The Astral 2000 monitor also has larger scale data manipulation facilities which include memory check sum calculation, "block insert" for temporary program patches, a command which can be used to calculate relative branch offsets, and patch points for the software interrupt, nonmaskable interrupt or maskable interrupt requirements of applications systems and programs.

High Level Language Facilities

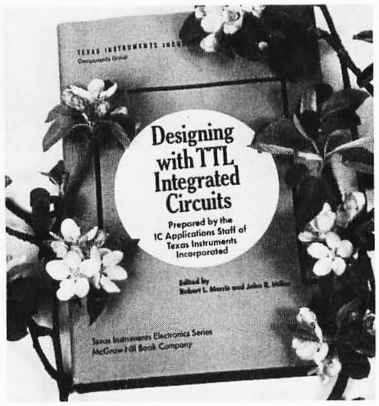
A major concern of most users is the need for high level language facility. In the Astral 2000 case, one of the strong points in the descriptive literature is its 8 K extended BASIC implementation. This is part of a standard $35 software package which is available to the purchaser of the system. The descriptive literature of Astral BASIC implies that it will prove quite useful, whether the application is business processing, games hacking, or a host of other small computer uses. Here is a summary list of the points found in the Astral BASIC which supplement the usual features of any good extended BASIC.

- Floating point arithmetic with user selectable precisions of 6, 9, 13 or 16 decimal digits. (Trade execution time off against precision of calculation.)
- A unique DO statement references groups of statements by number to provide a simple form of subroutine call not usually found in BASIC.
- Full statement trace facility.
- Variable length character strings of unlimited (ie: 2**16) length.
- Substring search function for character strings.
- Complete line oriented editing facility including a renumbering facility which alters all GO TOs.
- Formatted output with PRINT USING, an essential for practical applications programming.

Miscellaneous

Rounding out the details of the Astral 2000, the standard version includes an audio interface for magnetic tape recording, a real time clock based upon a standard 24 hour clock chip accessed through PIAs and level shifters, and a centralized power supply with 5 V at 12 A. Prices for several of the component modules are printed in the caption for photo 2.■

Still More BYTE's Books





•DESIGNING WITH TTL INTEGRATED CIRCUITS by the Components Group, Texas Instruments Inc. Edited by Robert L Morris and John R Miller.

People often ask questions like "Where do I get basic information on hardware design?" One answer is in **"Designing With TTL Integrated Circuits."**

This book, published by McGraw Hill in 1971, is a fundamental starting point for any person designing peripherals and custom logic employing TTL integrated circuits. While its publication date precludes any reference to the later additions to the TTL 7400 series of components found in the **Data Books,** it is nevertheless the source of a wealth of ideas on TTL integrated circuits and design of logic with this family of circuits.

What is fanout? You may have heard this term mentioned at computer club meetings or in advertisements for circuitry, or in articles in BYTE. You can find out background information on the calculation of fanouts by reading the chapter on **Circuit Analysis and Characteristics of Series 54/74.**

Worried about noise, shielding, grounding, decoupling, cross talk and transmission line effect? (Or, more properly, did you know you should worry about these effects in certain circumstances?) Find out about general precautions and background information by reading the chapter on **Noise Considerations.**

The chapter on **Combinatorial Logic Design** gives 53 pages of background information on Boolean algebra and practical representations of logic in the form of SSI gates. The chapter includes a description of Karnaugh mapping techniques and the minimization of logic. From combinatorial design, the book progresses into **Flip Flops,** including background information on the workings of these devices, and fairly detailed descriptions of the uses and applications of these devices including synchronization of asynchronous signals, shift registers, flip flop one shots, etc. Then the book returns to static combinatorial logic with its description of the **Decoders** available in the 7400 line as it stood in 1970-1971.

A chapter on **Arithmetic Elements** gives fundamental descriptions of binary arithmetic, diagrams of the basic gate configurations for combinatorial logic adders, and a section on number representations for use in computers. Much of the material in this section is dated, due to the fact that the later 74181 series of multiple function arithmetic units had not yet appeared when the book was written. But for a background on arithmetic operations implemented with the simpler 7483 circuits, this chapter is ideal. A chapter on **Counters** and a chapter on **Shift Registers** complete the detail logic sections. The book is closed out by a chapter on miscellaneous **Other Applications** including a simple binary multiplier, a 12 hour digital clock and a modulo-360 adder.

The most important use of this book is its value as an introduction to TTL logic. By reading and studying it, you will begin to understand the ways in which SSI and MSI TTL gates can be utilized in your own experimental logic designs. After studying this text, you should be able to make much more sense out of the technical information summaries typically published as specifications sheets and data catalogs.

Order your copy today from BYTE's Books, $24.

•MICROCOMPUTER DESIGN by Donald P Martin, Martin Research. Edited and Published by Kerry S Berland, Martin Research.

Purchase your copy of the definitive source for circuitry and hardware design information on the 8008 and 8080 computers today.

Even Intel, the originator of the microprocessor revolution, is hard put to compete with the wealth of information found in Martin Research's new second edition of **Microcomputer Design.** This is the book which was originally published as an expensive (but quite practical) engineering report in loose leaf form, at about the time the microprocessor technology was first catching on in the form of the 8008. This 388 page second edition of the manual is loaded with detailed information on how to build and use computers based on the 8008 and 8080.

But even if you do not intend to use the 8008 or 8080, the practical pointers on digital logic design, peripherals and applications of hardware techniques will more than justify the new low price of $25 for this handbook. **Microcomputer Design** is a must for 8008 owners and 8080 owners who want to truly understand how their processors process.

Microcomputer Design is complete with numerous illustrations, tables and diagrams, plus reprints of the specifications sheets for the Intel processors. There are numerous practical examples of circuitry and many complete computer designs ranging from "minimal microcomputers" to a full blown 8080 processor.

Order your copy today, $25 from BYTE's Books

BYTE PETERBOROUGH, NH 03458

___ Designing With TTL Integrated Circuits $24
___ Microcomputer Design $25

Send to: Name ____________________
Address ____________________
City ____________ State ________ Zip ________

master charge

BANKAMERICARD

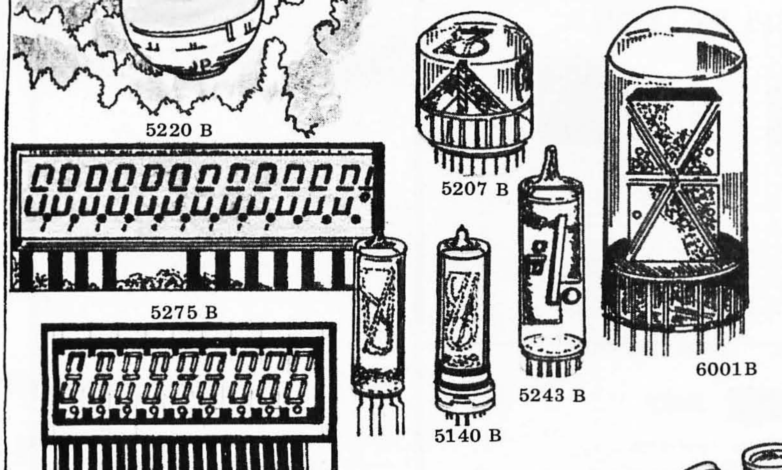
☐ Check enclosed
☐ Bill MC No. ____________ Exp. Date ________
☐ Bill BA No. ____________ Exp. Date ________
Signature ____________________

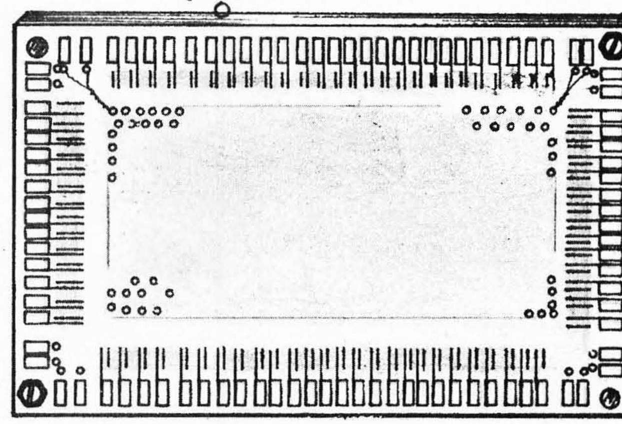
Please add 25 cents for each book, for postage and handling. Please allow six weeks for delivery.

DELTA ELECTRONICS CO.

POST OFFICE BOX 2 AMESBURY, MASS. 01913 PHONE (617)388-4705







CORE MEMORY BOARD

This memory board has 2 non-volatile magnetic core memory planes, each plane consisting of a 26 x 52 bit array; Total capacity is 2704 bits (1352x2). Four could be used to make a 1352 word memory, 8 bits wide. Because of its ability to retain data when power is removed, core memory is uniquely suited for bootstrap loaders, program development memore space, and whenever power failure is a major concern, such as in a security systems. 2 3/4 x 4 1/4 inches;
STOCK NO. 5446B $2.50 4/ 9.00

GAINT NIXIE TUBE ASSEMBLY
2 burroughs B7971 15 segment Alpha-Numerical tubes. 2.50 inch characters. 2 sockets mounted on driver board with 33 transistors. Perfect for large clocks and Alpha-Numerical displays. Data supplied 8 inches x 10 3/4 inches x 2 1/2 inches. 3 lbs.
STOCK NO . 6001 B $6.95 2/12.00

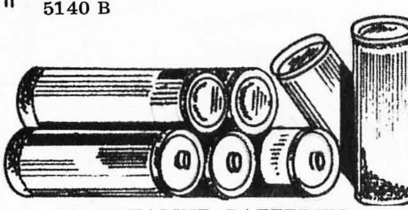
NIXIE TUBE
Left side display low-voltage read-out. Compatible with IC decoder or drivers. Fits low cost 9 pin sockets. 0.8 inch mounting
STOCK NO. 5243 B $1,25 5/ 1.00

END VIEW NUMITRONS
A 7 segment incandescent read-out tube. (5/8 characters) Requires a 9 pin Novar socket (supplied).It is a 4.5 volt 24 mA per segment. Can be filtered to any color. Data supplied.
STOCK NO. 5207 B $2.75 4/10.00 8/18.00

TUNG-SOL DIGIVAC READOUT
A 7 segment fluorescent read out tube (.36x.57 character) requires a 9 pinregular socket(supplied) 12 to 50 volts per segment. Readout,green
STOCK NO. 5140 B $2.65 2/ 5.00 6/14.00

SPERRY INFORMATION DISPLAY
Direct interfacing with MOS/LS. 9 digits, includes point and comma. Characters .25 inches intended for multiplex operation, complete data supplied
STOCK NO. 5275 B $2.95 2/ 5.00

BURROUGHS DISPLAYS
12 digit panel display contains comma's and periosd. Characters .25 inches ideal for any device requiring multipul digits. Compatable with DTL' TTL and C MOS. Neon glow discharge, orange in color. Socket and data supplied
STOCK NO. 5220 B $2.95 2/ 5.00



AA ALKALINE BATTERIES
AA 1.5v Penlight batteries are great for equipment and alike. Also advantages for the hobbiests. All batteries are guaranteed before shipment. Most all AA batteries are sold competitively for two (2) for $1.59..BUT DELTAS'.........
STOCK NO. 5443 B 4/$1.00...



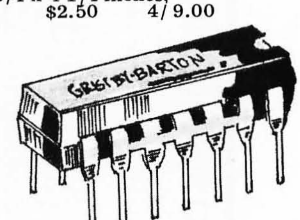
CRYSTAL

4.000,000 MHz CR-18A/U. Crystal can is .75x.75. The pins are .05 thk and the mounting spacing is .50. The crystal pin spacing is standard.
STOCK NO. 1001 B $3.95 2/7.00

CRYSTAL HOLDER
STOCK NO. 9169 B 561.00

GRIGSBY BARTON

DRY REED RELAY



Type G821 A SPST-14 pin, 24vdc 2150 ohms. Sheilded diode non-electrostatic sheild. They will fit standard 14-pin receptacles, they are designed for direct printed circuit board mounting where desirable. Internal clamp in circuit Bd mounting where it is desirable. Internal clamp diodes for protection of drivers by suppressing back EMF There is data available.

STOCK NO.. 1003 B $1.75 2/ 3.00

P.C. ROTARY SWITCHES
Half inch switches designed with only two moving parts. Mount directly on Printed Circuit Boards or panels. Positive detent, 36° indexing stops. 250 mA at 28vdc. 4,000 switching positions per position-contact resistance 0.05-ohm max. Operation temperature -65° C to - 150° C.

87-22-25 Knob Adjust 0.555 dia x 0.280 high
2 pole - 5 position.
STOCK NO. 1004 B $2.00 4/ 7.00



87-12-10 Screw driver Adjust 0.500 dia x 0.190 high
Single pole 10 position

STOCK NO. 1005 B $1.75 2/ 3.00

SEND FOR OUR LATEST FREE CATALOG. WE WELCOME MASTER CHARGE & BANK AMERICARD. WE MUST HAVE ALL THE NUMBERS ON THE CARD FOR PROCESSING. PLEASE INCLUDE SUFFICIENT POSTAGE (2 LBS MIN); EXCESS WILL BE REFUNDED.
MINIMUM ORDER $5.

7400N TTL

SN7400N*	.16	SN7459A	.25		
SN7401N	.16	SN7460N	.22	SN74154N*	1.00
SN7402N	.21	SN7470N	.45	SN74155N	.99
SN7403N	.16	SN7472N	.39	SN74156N	.99
SN7404N*	.18	SN7473N*	.37	SN74157N	.99
SN7405N	.24	SN7474N*	.32	SN74160N	1.25
SN7406N	.20	SN7475N*	.50	SN74161N	.99
SN7407N	.29	SN7476N*	.32	SN74163N*	.99
SN7408N	.25	SN7479N*	5.00	SN74164N	1.10
SN7409N	.25	SN7480N	.50	SN74165N	1.10
SN7410N*	.18	SN7482N	.98	SN74166N	1.25
SN7411N	.30	SN7483N	.70	SN74167N	5.50
SN7412N	.33	SN7485N	.89	SN74170N	2.10
SN7413N	.45	SN7486N	.39	SN74172N	8.95
SN7414N	.70	SN7488N	3.50	SN74173N	1.50
SN7416N	.35	SN7489N	2.25	SN74174N	1.25
SN7417N	.35	SN7490N*	.45	SN74175N	.99
SN7420N	.21	SN7491N	.75	SN74176N	.90
SN7421N	.33	SN7492N	.49	SN74177N	.90
SN7422N*	.49	SN7493N*	.49	SN74180N	.99
SN7423N	.37	SN7494N	.79	SN74181N	2.49
SN7425N	.29	SN7495N	.79	SN74182N	.95
SN7426N	.29	SN7496N	.89	SN74184N	1.95
SN7427N	.37	SN7497N*	4.00	SN74185N	2.20
SN7429N	.42	SN74100N*	1.00	SN74186N*	5.00
SN7430N	.26	SN74107N	.39	SN74187N	6.00
SN7432N	.31	SN74121N*	.39	SN74188N	3.95
SN7437N	.27	SN74122N	.39	SN74190N	1.19
SN7438N	.27	SN74123N*	.70	SN74191N	1.25
SN7439N	.25	SN74125N	.60	SN74192N*	.89
SN7440N	.15	SN74126N	.60	SN74193N*	.89
SN7441N	.89	SN74132N	1.09	SN74194N	1.25
SN7442N	.59	SN74136N	.95	SN74195N	.75
SN7443N	.75	SN74141N	1.15	SN74196N	1.25
SN7444N	.75	SN74142N*	4.00	SN74197N	.75
SN7445N	.75	SN74143N*	4.50	SN74198N	1.75
SN7446N	.81	SN74144N*	4.50	SN74199N	1.75
SN7447N*	.69	SN74145N	1.15	SN74200N	5.59
SN7448N	.79	SN74147N	2.35	SN74279N*	.90
SN7450N	.26	SN74148N	2.00	SN74251N	1.79
SN7451N	.27	SN74150N	1.00	SN74284N	6.00
SN7453N	.27	SN74151N	.79	SN74285N	6.00
SN7454N	.20	SN74153N	.89	SN74367N	.75

MANY OTHERS AVAILABLE ON REQUEST
20% Discount for 100 Combined 7400's

CMOS

CD4000	.25			74C04N	.75
CD4001	.25			74C10N	.65
CD4002	.25	CD4035	1.85	74C20N	.65
CD4006	2.50	CD4040	2.45	74C30N	.65
CD4007	.25	CD4042	1.90	74C42N	2.15
CD4009	.59	CD4044	1.50	74C73N	1.50
CD4010	.59	CD4046	2.51	74C74	1.15
CD4011	.25	CD4047	2.75	74C90N	3.00
CD4012	.25	CD4049	.79	74C95N	2.00
CD4013	.47	CD4050	.79	74C107N	1.25
CD4016	.56	CD4051	2.95	74C151	2.90
CD4017	1.35	CD4053	2.95	74C154	4.00
CD4019	.55	CD4060	3.25	74C157	2.15
CD4020	1.49	CD4066	1.75	74C160	3.25
CD4022	1.25	CD4069	.45	74C161	3.25
CD4023	.25	CD4071	.45	74C163	3.00
CD4024	1.50	CD4081	.45	74C164	3.25
CD4025	.25	CD4511	2.50	74C173	2.60
CD4027	.69	CD4518	2.50	74C193	2.75
CD4028	1.65	CD4566	3.00	74C195	2.75
CD4029	2.90	74C00N	.39	MC4044*	4.50
CD4030	.65	74C02N	.55	MC14016*	.56

LINEAR

LM300H	.80			LM1310N	2.95
LM301H	.35			LM1351N	1.65
LM301CN	.35			LM1414N	1.75
LM302H	.75			LM1458C	.65
LM304H	1.00	LM370N	1.1.5	LM1496N	.95
LM305H	.95	LM373N	3.25	LM1556V	1.85
LM307CN	.35	LM377N	4.00	LM2111N	1.95
LM308H	1.00	LM380N	1.39	LM2901N	2.95
LM308CN	1.00	LM390CN	1.05	LM3065N	.69
LM309H	1.10	LM381N	1.79	LM3900N	.55
LM309K	.99	LM382N	1.79	LM3905N	.60
LM310CN	1.15	NE501K	8.00	LM3909*	1.25
LM311H	.90	NE510A	6.00	LM5556N	1.85
LM311N	.90	NE531H	3.00	MC5558V	1.00
LM318CN	1.50	NE536T	6.00	LM7525N	.90
LM319N	1.30	NE540L	6.00	LM7535N	1.25
LM320K-5	1.35	NE550N	.79	8038B*	4.95
LM320K-5.2	1.35	NE555V*	.45	LM75450	.49
LM320K-12	1.35	NE560B*	5.00	75451CN	.39
LM320K-15	1.35	NE561B*	5.00	75452CN	.39
LM320T-5*	1.75	NE562B*	5.00	75453CN	.39
LM320T-5.2	1.75	NE565H*	1.25	75454CN	.39
LM320T-8*	1.75	NE565N*	1.75	75491CN	.79
LM320T-12*	1.75	NE566CN*	1.25	75492CN	.89
LM320T-15	1.75	NE567H*	1.95	75494CN	.89
LM320T-18*	1.75	NE567V*	1.50	**RCA LINEAR**	
LM320T-24	1.75	LM703CN	.45	CA3013	2.15
LM323K-5*	9.95	LM709H	.29	CA3032	2.56
LM324N	1.80	LM709N	.29	CA3035	2.48
LM339N	1.70	LM710N	.79	CA3039	1.35
LM340K-5	1.95	LM711N	.39	CA3046	1.30
LM340K-6	1.95	LM723H	.55	CA3059	3.25
LM340K-8	1.95	LM723H	.55	CA3060	3.25
LM340K-12	1.95	LM733N	1.00	CA3080	.85
LM340K-15	1.95	LM739N	1.00	CA3081	2.00
LM340K-18	1.95	LM741CH	.35	CA3082	2.00
LM340K-24	1.95	LM741CN	.35	CA3083	1.60
LM340T-5	1.75	LM74114N	.39	CA3086	.85
LM340T-6	1.75	LM747H	.79	CA3089	3.75
LM340T-8*	1.75	LM747N	.79	CA3091	10.20
LM340T-12	1.75	LM748H	.39	CA3123	2.15
LM340T-15	1.75	LM748N	.39	CA3130	1.39
LM340T-18*	1.75	LM1303N	.90	CA3140	1.25
LM340T-24	1.75	LM1304N	1.19	CA3600	1.75
LM350N	1.00	LM1305N	1.40	RC4194	5.95
LM351CN	.65	LM1307N	.85	RC4195	3.25

74LS00 TTL

74LS00	39			74LS139	1.95
74LS02	.39			74LS151	1.55
74LS03	.39	74LS74	.65	74LS153	1.89
74LS04	.45	74LS75	.79	74LS157	1.55
74LS05	.45	74LS76	.65	74LS162	2.25
74LS08	.39	74LS83	2.19	74LS163	2.25
74LS10	.39	74LS85	2.49	74LS164	2.25
74LS13	.79	74LS86	.65	74LS175	1.95
74LS14	2.19	74LS90	1.25	74LS181	3.69
74LS20	.39	74LS92	1.25	74LS190	2.85
74LS26	.49	74LS93	1.25	74LS191	2.85
74LS27	.45	74LS95	2.19	74LS192	2.85
74LS28	.49	74LS96	1.89	74LS193	2.85
74LS30	39	74LS107	65	74LS194	2.25
74LS32	.45	74LS109	.65	74LS195	2.25
74LS40	.49	74LS112	.65	74LS257	1.89
74LS51	.39	74LS132	1.55	74LS260	.55
74LS55	.39	74LS136	.65	74LS279	.79
74LS73	.65	74LS138	1.89	74LS670	5.95

DATA HANDBOOKS

7400	Pin-out & Description of 5400/7400 ICS	$2.95
CMOS	Pin-out & Description of 4000 Series ICS	$2.95
Linear	Pin-out & Functional Description	$2.95

ALL THREE HANDBOOKS $6.95

100 PER STRIP MOLEX PINS

Intended for use as an inexpensive substitute for IC sockets. Also perfect for use as board connectors and in subassemblies.

SPECIAL — 100/1.49 — 1000/12.00

CONSUMER ELECTRONICS

PONG — SINGLE GAME $55.00
SUPER PONG — 4 GAMES IN ONE $79.95

PONG



GAMES INCLUDED IN SUPER PONG ARE:

- PONG
- CATCH
- SUPER PONG
- HANDBALL

FEATURES OF PONG AND SUPER PONG

- Incremental speed on volleys increases excitement.
- Playing field adjusts to any size screen.
- Game appears in color or in black & white, depending on television set.
- Unmistakable "PONG" sound accompanies each volley.
- Digital scoring flashes on the screen between each point.
- 2 player challenge or Solitaire
- Hooks up simply to any model television set; the screen actually becomes the playing field.
- English and other techniques can be used to make any member of the family a Pong champion.
- Battery operated by 4 size "D" flashlight batteries included with the Unit.

AC Adaptor (Eliminates Batteries) **$9.95**

ATARI Innovative leisure

DISCRETE LEDS

.125" dia.

XC209Red	10/$1
XC209Green	4/$1
XC209Yellow	7/$1
XC209Orange	4/$1

.190" dia.

XC111Red	10/$1
XC111Green	4/$1
XC111Yellow	4/$1
XC111Orange	4/$1

.200" dia.

XC22Red	10/$1
XC22Green	4/$1
XC22Yellow	4/$1
XC22Orange	4/$1
SSL-22RT	4/$1

.185" dia.

XC526Red	10/$1
XC526Green	4/$1
XC526Yellow	4/$1
XC526Orange	4/$1
XC526Clear	4/$1

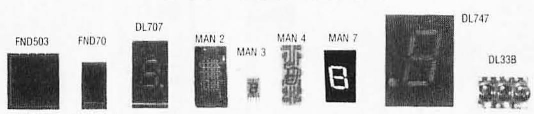
.200" dia.

XC556Red	10/$1
XC556Green	7/$1
XC556Yellow	7/$1
XC556Orange	7/$1
XC556Clear	7/$1

.085" dia.

MV50
.085" dia. Micro red LED
6/$1

DISPLAY LEDS



TYPE	POLARITY	HT		TYPE	POLARITY	HT	
MAN 1	Common Anode	.270	2.95	MAN 3620	Common Anode-orange	.300	1.75
MAN 2	5 x 7 Dot Matrix	.300	4.95	MAN 3640	Common Cathode-orange	.300	1.75
MAN 3	Common Cathode	.125	.39	MAN 4710	Common Anode-Red	.400	1.95
MAN 4	Common Cathode	.187	1.95	DL701	Common Anode-red	.300	1.85
MAN 7	Common Anode	.300	1.25	DL707	Common Anode	.300	1.50
MAN 7G	Common Anode-green	.300	1.95	DL 728	Common Cathode	.500	2.95
MAN 7Y	Common Anode-yellow	.300	1.95	DL 747	Common Anode	.600	2.25
MAN 52	Common Anode-green	.300	1.75	DL 750	Common Cathode	.600	2.49
MAN 64	Common Anode-red	.400	1.75	DL 338	Common Cathode	.110	.99
MAN 72	Common Anode	.300	1.25	FND70	Common Cathode	.250	.75
MAN 74	Common Cathode	.300	1.50	FND503	Common Cathode	.500	1.29
MAN 82	Common Anode-yellow	.300	1.75	FND507	Common Anode	.500	1.29
MAN 84	Common Cathode-yellow	.300	1.75				

IC SOLDERTAIL — LOW PROFILE (TIN) SOCKETS

	1-24	25-49	50-100		1-24	25-49	50-100
8 pin	$.17	16	.15	24 pin	$.38	.37	.36
14 pin	.20	.19	18	28 pin	.45	.44	.43
16 pin	.22	.21	20	36 pin	.60	.59	.58
18 pin	.29	.28	.27	40 pin	.63	.62	.61
22 pin	.37	.36	35				

SOLDERTAIL STANDARD (TIN)

14 pin	$.27	.25	.24	28 pin	$ .99	.90	.81
16 pin	.30	.27	.25	36 pin	1.39	1.26	1.15
18 pin	.35	.32	.30	40 pin	1.59	1.45	1.30
24 pin	.49	.45	.42				

SOLDERTAIL STANDARD (GOLD)

8 pin	$.30	.27	.24	24 pin	$ .70	.63	.57
14 pin	.35	.32	.29	28 pin	1.10	1.00	.90
16 pin	.38	.35	.32	36 pin	1.75	1.40	1.26
18 pin	.52	.47	.43	40 pin	1.75	1.59	1.45

WIRE WRAP SOCKETS (GOLD) LEVEL #3

10 pin	$.45	.41	.37	24 pin	$1.05	.95	.85
14 pin	.39	.38	.37	28 pin	1.40	1.25	1.10
16 pin	.43	.42	.41	36 pin	1.59	1.45	1.30
18 pin	.75	68	.62	40 pin	1.75	1.55	1.40

50 PCS. RESISTOR ASSORTMENTS $1.75 PER ASST.

		10 OHM	12 OHM	15 OHM	18 OHM	22 OHM	
ASST. 1	5 ea.	27 OHM	33 OHM	39 OHM	47 OHM	56 OHM	1/4 WATT 5% = 50 PCS.
		68 OHM	82 OHM	100 OHM	120 OHM	150 OHM	
ASST. 2	5 ea.	180 OHM	220 OHM	270 OHM	330 OHM	390 OHM	1/4 WATT 5% = 50 PCS.
		470 OHM	560 OHM	680 OHM	820 OHM	1K	
ASST. 3	5 ea.	1.2K	1.5K	1.8K	2.2K	2.7K	1/4 WATT 5% = 50 PCS.
		3.3K	3.9K	4.7K	5.6K	6.8K	
ASST. 4	5 ea.	8.2K	10K	12K	15K	18K	1/4 WATT 5% = 50 PCS.
		22K	27K	33K	39K	47K	
ASST. 5	5 ea.	56K	68K	82K	100K	120K	1/4 WATT 5% = 50 PCS.
		150K	180K	220K	270K	330K	
ASST. 6	5 ea.	390K	470K	560K	680K	820K	1/4 WATT 5% = 50 PCS.
		1M	1.2M	1.5M	1.8M	2.2M	
ASST. 7	5 ea.	2.7M	3.3M	3.9M	4.7M	5.6M	1/4 WATT 5% = 50PCS.
ASST. 8R		All Seven Resistor Assortment					$10.95 ea.

5-25 PCS: .05 ea. 30-95 PCS .04 ea. 100-495 PCS .03 ea. 500-995 .0275 ea

SINGLE TURN 3/8" CERMET POTENTIOMETER

- Resistance Tolerance ± 20%
- High Power −0.5 Watt at 70°C
- Wide Operating Temp. Range −55°C to +125°C

$.89

STANDARD RESISTANCE VALUES

50Ω 100Ω 500Ω 1K 2K 5K 10K 20K 50K 100K 200K 500K 1 MEG

All Advertised Prices Good For This Month Only — November
Satisfaction Guaranteed. $5.00 Min. Order. U.S. Funds.
California Residents — Add 6% Sales Tax — Data Sheets 25¢ each
Send a 24¢ Stamp (postage) for a Free 1977 Catalog

NOW OPEN SATURDAYS

JAMES

1021 HOWARD AVE., SAN CARLOS, CA. 94070
PHONE ORDERS WELCOME — (415) 592-8097

MINIATURE TOGGLE

DPDT				JMT	1	2-9	10-29	30-99
	ON	OFF	ON	221	2.95	2.55	1.87	1.70
	ON	NONE	ON	223	2.95	2.15	1.58	1.43
	ON	OFF	ON	121	2.35	1.95	1.43	1.30
	ON	NONE	ON	123	2.05	1.65	1.21	1.10



PUSH BUTTON

SPDT – SERIES PB, PB1

Maintained Action Switch Push To Make Contact And Push Again To Reserve Action

Momentary Action Switch Normally Open or Normally Closed (For N.C circuit use terminals 2 & 1; for N.O. 2 & 3; for N.C and N.O. 1,2,&3

PB-123 $1.75

PB-126 $1.75

5 AMP RATINGS



THUMBWHEEL SWITCHES

THUMBWHEEL SWITCH ONLY

Part No.	Description	Price
SF 12	Single Pole 10 Position	$2.50
SR 12	Decimal	3.00
SF 21	10 Position BCD only	2.50
SR 21		3.00

Ordering: Order desired switch or switches and add necessary accessories for your particular application.

SF – Front Mount SR – Rear Mount

SERIES SF Front Mount Assembly



SERIES SR Rear Mount Assembly



ACCESSORIES

Part No.	Description	Price
SF EP	End Plate (Pair)	.50
SF DP	Divider Plate (each)	.40
SF BB	Blank Body (each)	.40
SF HB	Half Body (each)	.40

ACCESSORIES

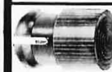
Part No.	Description	Price
SR EP	End Plates (pair)	$ .50
SR DP	Divider Plate (each)	.40
SR BB	Blank Body (each)	.40
SR HB	Half Body (each)	.40

DIP SWITCH

These switches feature seven SPST slide switches in a molded dip. They are ideally suited for microprocessor applications.

$1.95

CLIPLITE !! 4/$1.00



NEW LED MOUNTING SYSTEM

- CLIPLITE mounts from the front of the Panel in a .250 hole on 3/8" centers. Panel thickness from 1/16" to 1/8".
- CLIPLITE equalizes and increases the brightness of commonly used wide beam LEDS.
- CLIPLITE is to be used with XC556 LEDS.
- Specify colors–red, green, amber, or yellow when ordering.

ZENERS—DIODES—RECTIFIERS

TYPE	VOLTS	W	PRICE	TYPE	VOLTS	W	PRICE
IN746	3.3	400m	4/1.00	1N4005*	600 PIV	1 AMP	10/1.00
1N751A	5.1	400m	4/1.00	1N4006*	800 PIV	1 AMP	10/1.00
1N752	5.6	400m	4/1.00	1N4007*	1000 PIV	1 AMP	10/1.00
1N753	6.2	400m	4/1.00	1N3600	50	200m	6/1.00
1N754	6.8	400m	4/1.00	1N4148	75	10m	15/1.00
1N959	8.2	400m	8/1.00	1N4154	35	10m	12/1.00
1N965B	15	400m	4/1.00	1N4305	75	25M	20/1.00
1N5232	5.6	500m	28	1N4734	5.6	1w	28
1N5234	6.2	500m	28	1N4735	6.2	1w	28
1N5235	6.8	500m	28	1N4736	6.8	1w	28
1N5236	7.5	500m	28	1N4738	8.2	1w	28
1N456	25	40m	6/1.00	1N4742	12	1w	28
1N458	150	7m	6/1.00	1N4744	15	1w	28
IN485A	180	10m	5/1.00	1N1183	50 PIV	35 AMP	1.60
1N4001	50 PIV	1 AMP	12/1.00	1N1184	100 PIV	35 AMP	1.70
1N4002	100 PIV	1 AMP	12/1.00	IN1185	150 PIV	35 AMP	1.70
1N4003	200 PIV	1 AMP	12/1.00	IN1186	200 PIV	35 AMP	1.80
1N4004	400 PIV	1 AMP	12/1.00	1N1188	400 PIV	35 AMP	3.00

SCR AND FW BRIDGE RECTIFIERS

C36D	15A @ 400V	SCR	$1.95
C38M	35A @ 200V	SCR	1.95
2N2328	1.6A @ 200V	SCR	.50
MDA 980-1	25A @ 50V	FW BRIDGE REC.	1.95
MDA 980-3	25A@ 200V	FW BRIDGE REC.	1.95

TRANSISTORS

MPS A05	5/$1.00			PN4249	4/$1.00
MPS A06	5/$1.00			PN4250	4/$1.00
2N2219A	3/$1.00	PN3567	3/$1.00	2N4400	4/$1.00
2N2221	4/$1.00	PN3568	4/$1.00	2N4401	4/$1.00
2N2222A	5/$1.00	PN3569	4/$1.00	2N4402	4/$1.00
2N2369	5/$1.00	2N3704	5/$1.00	2N4403	4/$1.00
2N2369A	4/$1.00	2N3705	5/$1.00	2N4409	5/$1.00
FN2415	5/$1.00	2N3706	5/$1.00	2N5086	4/$1.00
2N2484	4/$1.00	2N3707	5/$1.00	2N5087	4/$1.00
2N2906A	4/$1.00	2N3711	5/$1.00	2N5088	4/$1.00
2N2907A	5/$1.00	2N3724	$ .65	2N5089	4/$1.00
2N2925	5/$1.00	2N3725	$1.00	2N5129	5/$1.00
2N3053	2/$1.00	2N3903	5/$1.00	2N5138	5/$1.00
2N3055	$ .89	2N3904	4/$1.00	2N5139	5/$1.00
MJE3055	$1.00	2N3905	4/$1.00	2N5209	5/$1.00
MJE2955	$1.25	2N3906	4/$1.00	2N5951	5/$1.00
2N3392	5/$1.00	2N4013	3/$1.00	C106B1SCR	2/$1.00
2N3398	5/$1.00	2N4014	3/$1.00	2N5432	$2.00
		2N4123	10/$1.00	40673	1.75ea.

CAPACITOR CORNER

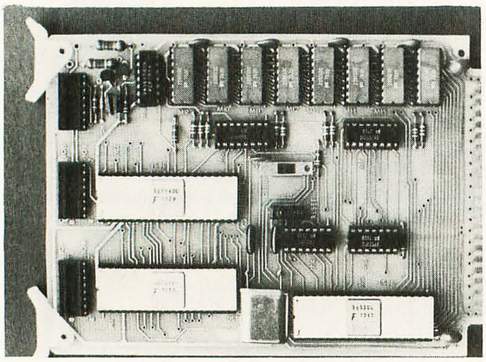
50 VOLT CERAMIC DISC CAPACITORS

	1-9	10-49	50-100		1-9	10-49	50-100
10 pf	.05	.04	.03	.001µF	.05	.04	.035
22 pf	.05	.04	.03	.0047µF	.05	.04	.035
47 pf	.05	.04	.03	.01µF	.05	.04	.035
100 pf	.05	.04	.03	.022µF	.06	.05	.04
220 pf	.05	.04	.03	.047µF	.06	.05	.04
470 pf	.05	.04	.035	.1µF	.12	.09	.075
100 VOLT MYLAR FILM CAPACITORS							
.001mf	.12	.10	.07	.022mf	.13	.11	.08
.0022	.12	.10	.07	.047mf	.21	.17	.13
.0047mf	.12	.10	.07	.1mf	.27	.23	.17
.01mf	.12	.10	.07	.22mf	.33	.27	.22
+20% DIPPED TANTALUMS (SOLID) CAPACITORS							
.1/35V	.28	.23	.17	1.5/35V	.30	.26	.21
.15/35V	.28	.23	.17	2.2/25V	.31	.27	.22
.22/35V	.28	.23	.17	3.3/25V	.31	.27	.22
.33/35V	.28	.23	.17	4.7/25V	.32	.28	.23
.47/35V	.28	.23	.17	6.8/25V	.36	.31	.25
.68/35V	.28	.23	.17	10/25V	.40	.35	.29
1.0/35V	.28	.23	.17	15/25V	.63	.50	.40
MINIATURE ALUMINUM ELECTROLYTIC CAPACITORS							
Axial Lead				**Radial Lead**			
.47/50V	.15	.13	.10	.47/25V	.15	.13	.10
1.0/50V	.16	.14	.11	.47/50V	.16	.14	.11
3.3/50V	.15	.13	.10	1.0/16V	.15	.13	.10
4.7/25V	.16	.14	.12	1.0/25V	.16	.14	.11
10/25V	.15	.13	.10	1.0/50V	.16	.14	.11
10/50V	.16	.14	.12	4.7/16V	.15	.13	.10
22/25V	.17	.15	.12	4.7/25V	.15	.13	.10
22/50V	.24	.20	.18	4.7/50V	.16	.14	.11
47/25V	.19	.17	.15	10/16V	.14	.12	.09
47/50V	.25	.21	.19	10/25V	.15	.13	.10
100/25V	.24	.20	.18	10/50V	.16	.14	.12
100/50V	.35	.30	.28	47/50V	.24	.21	.19
220/25V	.32	.28	.25	100/16V	.19	.15	.14
220/50V	.45	.41	.38	100/25V	.24	.20	.18
470/25V	.33	.29	.27	100/50V	.35	.30	.28
1000/16V	.55	.50	.45	220/16V	.23	.17	.16
2200/16V	.70	.62	.55	470/25V	.31	.28	.26

What's New?

A World's Smallest Microcomputer System?

The manufacturer of this card, the Model 710A, claims that this F8 computer card is "the world's smallest complete 8 bit microcomputer." The card comes fully assembled and tested and measures 4.5 by 6.5 inches



(11.4 cm by 15.2 cm) with a standard double sided 22 pin edge connector. The term "complete" has the following semantics according to the Pronetics Corp press release:

- F8 Microprocessor with 2 μs instruction time.
- 1 K byte programmable memory.
- 1 K bytes programmable memory.
- 32 bidirectional latched single bit IO ports.
- 2 independent external interrupt inputs.
- 2 independent programmable interval timers.
- Flexible terminal interface.
- Physical size fits standard card cage products.

At $159, completely assembled and tested, the F8 Model 710A looks like a fair fulfillment of the adjective "complete." Individuals looking for a dedicated firmware controller, a starting point for an F8 general purpose system, or a computer for use in educational or laboratory programs should find this board to be a useful option. The manufacturer is Pronetics Corp, 6431 Preston Crest, Dallas TX 75230, (214) 270-8626.■

Updates to the 9900 Line

Texas Instruments has announced additions to the TMS9900 microcomputer product family in a release dated July 20. A short summary of the additions, which should show up in personal computing end products in the next half year or so, is as follows:

- TMS9980: This is a new MOS version of the TMS9900, packaged as a standard 40 pin dual in line package with a physical description similar to most 8 bit microprocessors: 8 bit data bus, 16 bit address bus, etc. But there is a difference. If readers will recall Robert Baker's article on the TMS9900 chip in the April 1976 BYTE, the internal architecture of a 9900 is that of a 16 bit minicomputer with hardware multiply and divide, 16 general purpose register like work areas, etc. The same instruction set is executed by the 9980 with a slight drop in speed of execution due to the fact that byte addressing is now done in the external world with an 8 bit wide memory and peripherals data path instead of a 16 bit data path. This product is to be sampled late in 1976, and we expect at least one manufacturer to put this on an Altair compatible plug in board for the personal computing market. The hardware of the 9980 features an onchip oscillator and clock generator, so the breadboarding and production of hardware with this chip should prove convenient.
- TMS9904 is a 4 phase clock generator and driver for use with the TMS9900 computer circuit, using a crystal controlled input.
- TMS9901 is a programmable systems

S. D. SALES Co. P.O. BOX 28810-B • DALLAS, TEXAS 75228

4K LOW POWER RAM BOARD KIT

Imsai and Altair 8080 plug in compatible. Uses low power static 21L02–1 500 ns. RAM's, which are included. Fully buffered, drastically reduced power consumption, on board regulated, all sockets and parts included. Premium quality plated thru PC Board.

THE WHOLE WORKS

$89.95

---

8 Digit LED
"METRIC MASTER"
$19.95

"RAPID MAN – 12"
$29.95

$1,000,000 CALCULATOR PURCHASE!
We bought the entire stock of a major manufacturer. New, guaranteed units.

Five functions PLUS complete Metric Conversion functions. Rechargeable batteries. Small, hand held size. With AC charger.

12 Digit – Desk Top Style. Sturdy design. With memory and four complete functions. Big, bright display.

ALARM CLOCK KIT SIX DIGIT LED

Thousands of hobbyists have bought and built our original clock kit and were completely satisfied. But we have received many requests for an alarm clock kit with the same value and quality that you have come to expect from S. D. So, here it is!

THE KIT INCLUDES:
1 Mostek 50252 Alarm Clock Chip
6 Hewlett Packard .30 in. common cathode readouts
15 NPN Driver Transistors
2 Switches for time set
2 Slide Switches for alarm set and enable
1 Filter Cap
4 IN4002 Rectifiers
1 IN914 Diode
1 .01 Disc Cap
15 Resistors
1 Speaker for alarm
1 LED lamp for PM indicator

$9.95

PCB — $3.00
XFMR — $1.50

60 HZ CRYSTAL TIME BASE – FOR DIGITAL CLOCKS
S. D. SALES EXCLUSIVE!

$5.95

KIT FEATURES:
A. 60 hz output with accuracy comparable to a digital watch
B. Directly interfaces with all MOS Clock chips
C. Super low power consumption (1.5 Ma typ.)
D. Uses latest MOS 17 stage divider IC
E. Eliminates forever the problem of AC line glitches
F. Perfect for cars, boats, campers, or even for portable clocks at ham field days. BUY TWO FOR $10.00!
G. Small size, can be used in existing enclosures.

KIT INCLUDES CRYSTAL, DIVIDER IC, PC BOARD PLUS ALL OTHER NECESSARY PARTS AND SPECS.

MOTOROLA RTL IC'S

Brand new, factory prime. Hard to find, but still used in a variety of projects. (See the RTL Cookbook by Howard W. Sams.)

MC724P — 59c	MC780P — 89C	MC791P — 69c
MC725P — 59c	MC785P — 49c	MC792P — 59c
MC764P — 49c	MC787P — 89c	MC799P — 59c
MC767P — 69c	MC788P — 49c	MC970P — 89c
MC771P — 49c	MC789P — 59c	MC9709P — 69c
MC775P — 89c	MC790P — 89c	MC9760P — 69c

TTL INTEGRATED CIRCUITS

7400 — 19c	7430 — 19c	7476 — 35c	74153 — 75c
7402 — 19c	7432 — 34c	7480 — 49c	74154 — 1.00
74L04 — 29c	7437 — 39c	7483 — 95c	74157 — 75c
74S04 — 44c	7438 — 39c	7485 — 95c	74161 — 95c
7404 — 19c	7440 — 19c	7586 — 45c	74164 — 1.10
7406 — 29c	7447 — 85c	7490 — 65c	74165 — 1.10
7408 — 19c	7448 — 85c	7492 — 75c	74174 — 95c
7410 — 19c	7451 — 19c	7495 — 75c	74181 — 2.50
7411 — 29c	7453 — 19c	7496 — 89c	74191 — 1.25
7413 — 50c	7473 — 39c	74121 — 38c	74192 — 1.25
7420 — 19c	7474 — 35c	74123 — 65c	74193 — 1.00
	7575 — 35c	74141 — 75c	74195 — 69c

"CUBO" –DIGITAL ALARM CLOCK CUBE
A PERFECT GIFT – NOT A KIT!



The CUBO Alarm Clock mfg. by Corvus, division of MOSTEK CORP. Originally sold for $49.95. We bought out their entire inventory. All new, individually gift boxed. Mini size (2½" cube) with maxi performance.

FEATURES:
A. 4 Digit H.P. Bright Display
B. "Second Hand" LED Activity Indicator
C. 24 Hour Alarm; 12 hour real time format
D. 10 Minute Snooze
E. Auto Display Dimming — Adjusts to Ambient Light
F. AM/PM Indicator
G. Power Failure Indicator
Uses Famous Mostek MK50250 MOS LSI IC

$14.95 EACH (Add 75c P.&H.)

1000 MFD FILTER CAPS



Rated 35 WVDC. Upright style with P. C. leads. Most popular value for hobbyists. Compare at up to $1.19 each from franchise type electronic parts stores.
S.D. SPECIAL 4 for $1.

SLIDE SWITCH ASSORTMENT



Our best seller. Includes miniture and standard sizes, single and multi-position units. All new, first quality, name brand Try one package and you'll reorder more. **SPECIAL – 12/$1.**

RESISTOR ASSORTMENT



1/4 W 5% and 10%. PC leads. A good mix of values. **200/$2.**

MOTOROLA POWER DARLINGTON
Back in Stock!

Like MJ3001. NPN 80V. 10A. HFE 6000 TYP. TO–3 case. We include a free 723 C volt reg. with schematic for power supply. **SPECIAL – $1.99**

WESTERN DIGITAL UART

No. TR1602B. 40 pin DIP.
This is a very powerful and popular part.
NEW – $6.95 with data
LIMITED QUANTITY

74S200
256 Bit High Speed RAM
Same as 82S16
$3.95

FAIRCHILD BIG LED READOUTS

A big .50 inch easy to read character. Now available in either common anode or common cathode. Take your pick. Super low current drain, only 5 MA per segment typical. **YOUR CHOICE: 6 for $7.50**

FND – 510 Common Anode $1.50 ea.
FND – 503 Common Cathode $1.50 ea.

MICROPROCESSORS & THINGS
By AMD

8080A	Better than Intel	$24.00
8008-1	High Speed 8008	12.50
8212	I/0 Port	3.50
8224	Clock Generator	4.95
2513	Character Generator	10.00

1K PROM BACK IN STOCK!
82S129. 256X4. Bipolar, 50 NS. FAST. WITH SPECS.
$3.95

8T97B
Hex Tri-State Buffer. Back in stock.
$1.25

UP YOUR COMPUTER!
21L02–1 1K LOW POWER 500 NS STATIC RAM
TIME IS OF THE ESSENCE

And so is power. Not only are our RAM's faster than a speeding bullet but they are now very low power. We are pleased to offer prime new 21L02 — 1 low power and super fast RAM's, Allows you to STRETCH your power supply farther and at the same time keep the wait light off. **8 FOR $17.50**

CALL YOUR BANK AMERICARD OR MASTER CHARGE ORDER IN ON OUR CONTINENTAL UNITED STATES TOLL FREE WATTS: 1–800–527-3460

Texas Residents Call Collect
214/271-0022

SALE ON CUT LEAD SEMICONDUCTORS

Leads were cut for PCB insertion. Still very useable. All new, unused. Some House no.

1N914/1n4148 . . . . 100/$2.
1N4002 1 Amp 100 PIV . . . . 40/$1.
1N4745A 16V 1W Zener . . . . 20/$1.
EN2222 NPN Transistor . . . . 25/$1.
EN2907 PNP Transistor . . . . 25/$1.
2N3904 NPN Driver Xstr . . . . 25/$1.
2N3392 GE Pre-amp Xstr . . . . 25/$1.
C103Y SCR. 800MA 60V . . . . 10/$1.

C&K MINI TOGGLE SWITCH
No. 7103 SUB MINI SPDT Center OFF.
SPECIAL – 99c

TERMS:
Money Back Guarantee. No COD. Texas Residents add 5% tax. Add 5% of order for postage and handling. Orders under $10. add 75c. Foreign orders: US Funds ONLY!

S. D. SALES
P. O. Box 28810-B
Dallas, Texas 75228

ORDERS OVER $15. CHOOSE $1. FREE MERCHANDISE

interface for the TMS9900 or TMS9980, which interfaces directly to the processor's CRU port for interrupt priority, IO control and interval timing. It provides 15 individually maskable interrupt request lines or up to 16 programmable IO ports, and can be programmed as an interval timer with resolution from 21 μs to 699 ms.

- TMS9902 is the 9900 family's version of a UART. This circuit is customized to the TI CRU (Communications Register Unit) IO addressing scheme, and features data rates from 5 to 76,800 bits per second, character lengths from 5 to 8 bits, and 1, 1.5 or 2 stop bits per character with or without parity.
- TMS9903 is the synchronous communications port for the 9980 and 9900 computers. This is similar to what is sometimes called a "USART" and features a wide variety of options. It will handle most synchronous data transmission protocols, including the BiSync and IBM's new Synchronous Data Link Control discipline, at data rates from DC to 250,000 bits per second.

Samples have not yet been delivered on these items, and pricing is to be announced when samples are available. Inquiries from OEMs should be directed to Texas Instruments Inc, POB 5012, MS/84, Dallas TX 75222, attention "9900 Family". ■

Would You Like to Find Out About Hardware for Diagnosing and Testing Microprocessor Designs?

Hewlett-Packard makes a broad line of test equipment intended for industrial and commercial design and maintenance applications. A new applications note, AN 167-13, is now available, written by Bruce G Farly of Hewlett-Packard's Colorado Springs facility, describing some of the uses and applications of sophisticated testing equipment such as logic state analyzers and data generators. This 6 page applications note, entitled "The Role of Logic State Analyzers in Microprocessor Based Designs" is available for the asking by writing to: Inquiries Manager, Hewlett-Packard Co, 1501 Page Mill Rd, Palo Alto CA 94304.■

A New Graphics Option for OSI

Shown off at the Personal Computing 76 show in Atlantic City was the interface which produces these pictures. The manufacturer of the "Super Video Board" which generates 128 by 128 raster graphics is Ohio Scientific Instruments, 11679 Hayden St, Hiram OH 44234. This board, which plugs into the OSI Model 400 system, can be used, as in photo 1, to draw pictures with low to moderate resolution, and to display the output of a software "logic state analyzer" program, as in photo 2.■

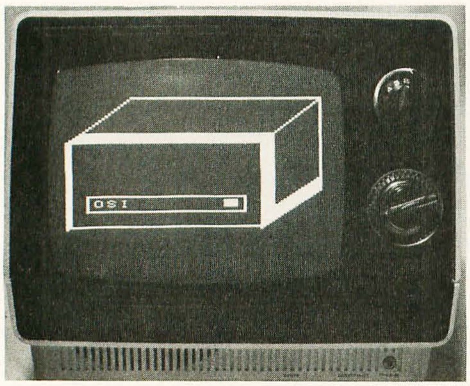


Photo 1.

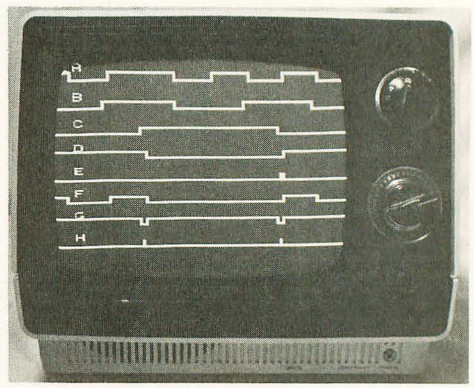
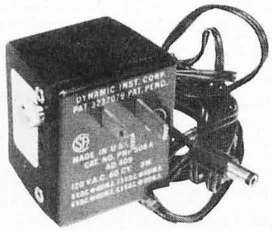


Photo 2.



UNIVERSAL POWER SUPPLY

A unique plug-in supply by Panasonic. Useful for calculators, small radios, charging many & various small NiCad batteries. Adjustment screw plug on the side changes output voltage to 4½, 6, 7½, or 9 volts DC at 100 MA. Output cord with plug, 6 ft long.

No. SP-143C $4.50 3/$12

POWER SUPPLY

LAMBDA 5VDC 74 AMP

LV-EE-5-OV $125.00

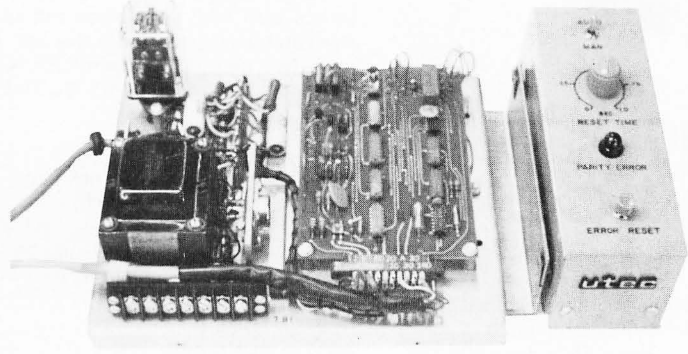
NJE 5/OUP-D5

5 VDC 32 AMP $75.00

CLOCK KIT $14.00

Includes all parts with MM5316 chip, etched & drilled PC board, transformer, everything except case.

SP-284 $14.00 each 2/$25.00



PARITY DETECTOR

New packaged, made for RCA, detects even or odd parity, baud rate 110, 150 or 134.46. Built-in logic supply for the IC's, operates from standard 115 vac. Control panel allows manual or automatic reset mode of operation. Aluminum enclosure (not shown), covers the electronics. TTY compatible.

Ship wgt. 10 lbs. $16.50

COMPUTER DISPLAY TUBE

New Sylvania 9 inch CRT, 85 degree deflection, with tinted faceplate. Same as used in Viatron systems (buy a spare). With complete specs.

Ship wgt. 5 lbs. $15.00

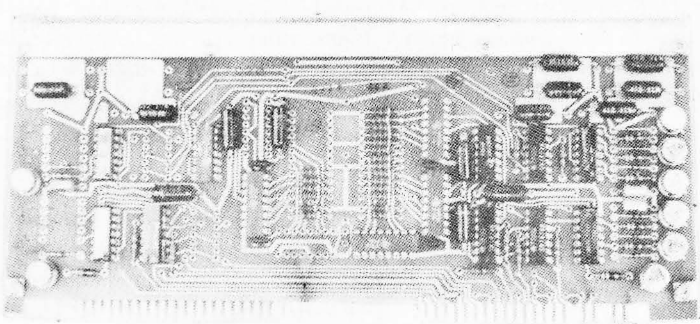
LINEAR by RCA, brand new, gold bond process

301	$ .60	747	$ .82	MM5314	$3.00
307	.52	748	.50	MM5316	3.00
324	1.80	1458	.96	7001	8.00
339A	1.60	3401	.80		
741	.50	555 timer	.60		

MEMORY SYSTEM $125.00

New memory system by Honeywell, small ... measures only 9x4x1 inches. 1024 core memory, 1024 words with 8,9,10 bits/word. Random access, with all logic, register, timing, control, core select and sense functions in one package. New, booklet of schematics and data. Looks like a good beginning for a mini-computer. Limited supply on hand.

Ship wgt 3 lbs. #SP-79 . . . . $125.00



---

3 Power supplies, transistorized & regulated. Made by TRANSISTOR DEVICES

15 volts DC 5 amps $25.00
30 volts DC 2 amps 25.00
15 volts DC 4.5 amps 25.00

COMPUTER GRADE LOGIC SUPPLY CAPS, BRAND NEW

47,000 Uf	25V	$2.00	ST	1,000	50	.90	AL
32,000	25	1.75	ST	3,300	35	1.25	AL
160,000	10	2.00	ST	1,600	20	.60	AL
66,000	10	2.00	ST	8,000	16	1.25	AL
1,000	60	.90	AL	500	6	.35	AL
2,000	55	1.00	AL				

"ST" screw top . . . . "AL" axial

Please add shipping cost on above.

Meshna

PHONE 617–595-2275 FREE CATALOG SP-8 NOW READY

MESHNA PO Bx 62 E. Lynn Mass. 01904

BYTE reader service

To get further information on the products advertised in BYTE, fill out the reader service card with your name and address. Then circle the appropriate numbers for the advertisers you select from this list. Add a 9 cent stamp to the card, then drop it in the mail. Not only do you gain information, but our advertisers are encouraged to use the marketplace provided by BYTE. This helps us bring you a bigger BYTE.

***Reader service inquiries not solicited. Correspond directly with company.**

Reader Service (?)

Subscribers, please bear with us. If you have not received a response to a BYTE reader service inquiry, the delay rests squarely upon us, not the advertisers. We have quite frankly been inundated with requests since putting reader service on reply cards, and recently got a bit behind in processing. Each card must be keypunched and run through a (somewhat traditional) data processing operation currently implemented on an IBM/360 Model 25 with 16 K bytes of memory, 2311 disk drives and an archaic tape operating system. As this note is written (September 10) the current reader service requests total 24,000 individual cards with multiple requests per card, and are being mailed to advertisers as mailing label printouts. ■

Feedback is what keeps a linear amplifier in line. Like a linear amplifier, BYTE can use a bit of feedback. The BOMB analysis is done once a month to provide encouragement to authors and some formal feedback on how readers appreciate articles. BYTE pays the winning author a $50 bonus, so you can encourage the authors you like by voting your preferences. Remember that with few exceptions BYTE authors are just readers who have sat down at their typewriters to tell a story about what they've done or what they know about some aspect of this technology.

BOMB: BYTE's Ongoing Monitor Box

PAGE	ARTICLE	LIKED: LEAST … BEST
6	Rosner: It's More Fun Than Crayons	0 1 2 3 4 5 6 7 8 9 10
20	Arnold: What Is APL?	0 1 2 3 4 5 6 7 8 9 10
26	Nelson: Beer Budget Graphics Interface	0 1 2 3 4 5 6 7 8 9 10
32	Buschbach: Add This Graphics Display . . .	0 1 2 3 4 5 6 7 8 9 10
40	Burhans: A Tip for Using Wiring Pencils	0 1 2 3 4 5 6 7 8 9 10
42	Deres: An Enterprising Display Device	0 1 2 3 4 5 6 7 8 9 10
56	Rampil: Some Graphics Background . . .	0 1 2 3 4 5 6 7 8 9 10
78	Ciarcia: A Real Eye Opener	0 1 2 3 4 5 6 7 8 9 10
106	Anderson: Video Display Terminal	0 1 2 3 4 5 6 7 8 9 10

BOMB Bonus to Burt

The winner of the August BOMB survey and a $50 bonus was Burt Hashizume, who wrote "Microprocessor Update: Zilog Z80." The runners-up for August were Robert Suding for his article on the Digital Group TV readout device, and Terry Steeden who wrote "What's an I^2L (I squared L)."

Remember that your preferences count in the BOMB evaluation. It is a way of giving direct feedback to authors about the content of their articles, and it gives us a real world basis for judgment of which articles to push into print the fastest. ■

SPHERE OFFERS:
A Programmable Microprocessor Controller!

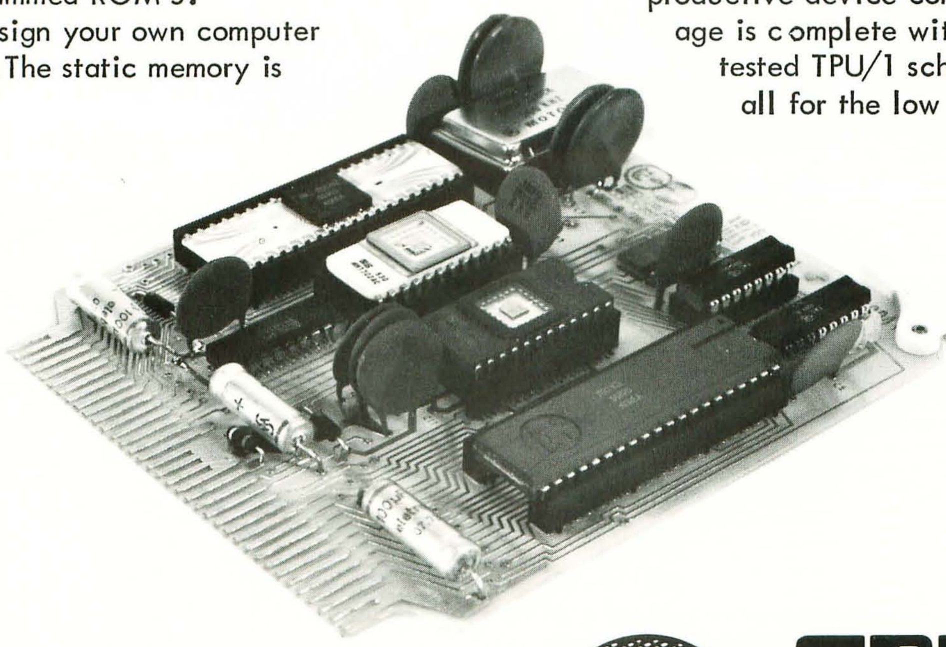
$194.00

The Sphere TPU/1 micro computer on a card designed for use in control enviroments; whether it be personal, Business, Idustrial, or Engineering. TPU is an expandable CPU on a 4.4x4.9 inch edge connector board. With the Motorola 6800 microprocessor a 6820 parallel interface 256 or 512 bytes of EPROM 2048 bits of static RAM, and a crystal controlled clock driver. Available on the 72 pin edge connector are all the necessary signals to interface the TPU/1 to almost any device. You can program the read only memory with special instructions to meet your needs, or buy our programmed ROM'S.

Build or design your own computer configuration. The static memory is expandable to 32k and you can control teletype or other I/O devices. Options include keyboard and 16 x 32 character CRT display module.

The TPU allows ease of interfacing to virtually any device which needs to be intelligently monitored or controlled. Whether need a personal computer to handle your household budget or to use in your next electronics project, this is the low cost unit which will do the job for you.

This is not a toy! It was designed for use in OEM applications. It is a reliable and productive device controller. The package is complete with assembled and tested TPU/1 schematics, manuals; all for the low price of $194 prepaid.



Order yours today and use the Sphere TPU/1 in your next electronics project. I want ______ quanity. Enclose BAC, MC or check. Allow up to 30 days for delivery.

SPHERE
CORPORATION
940 North 400 East
North Salt Lake, Utah 84054
(801) 292-8466



Measuring just 11" wide x 11" deep x 5" high, and weighing a mere 7 pounds, the Altair™ 680b is a complete, general-purpose computer.

The secret to this revolutionary, small computer is its CPU board. This double-sided board fits along the bottom of the Altair case and plugs directly into the front panel board. It contains the new 6800 microprocessor, 1,024 bytes of RAM memory, a 256 byte PROM monitor, provisions for 768 bytes of additional PROM or ROM, and a single Interface port with a Motorola ACIA serial interface adapter which can be configured either RS-232 or TTY. A five level Baudot interface option is also available.

The Altair 680b can be programmed from front panel switches, or it can be interfaced to a video display terminal, or teletypewriter. Three additional circuit boards can be plugged inside the Altair 680b for further memory and interface expansion. The first of these boards is a 16K static RAM memory board.

Software already developed includes Altair 680 BASIC with all the features of the 8K BASIC previously developed for the Altair 8800. These include Boolean operators, the ability to read or write a byte from any I/O port or memory location, multiple statements per line, and the ability to interrupt program execution and then continue after the examination of variable values. This software takes only 6.8K bytes of memory space and a copy is **included free with the purchase of the Altair 680 16K memory board.**

Other software includes a resident two pass assembler. The Altair 680b is also compatible with Motorola 6800 software.

The Altair 680b is ideal for hobbyists who want a powerful computer system at an economic price. Altair 680b owners qualify for membership in the Altair Users Group, and like other Altair owners, they receive a complimentary subscription to **Computer Notes** and complete factory support.

PRICES:

Altair 680b kit with complete, easy-to-understand assembly manual, operator's manual, and programming manual $466
Assembled Altair 680b $625
Altair 680b Turnkey model kit 395
Expander Card 680MB (required to expand 680) .... $ 24
Altair 680BSM 16K static RAM board kit with 680 BASIC $685
Altair 680 BASIC when purchased separately $200
Baudot option .... $ 42

MAIL THIS COUPON TODAY

☐ Enclosed is a check for $________
☐ BankAmericard #________ or Master Charge #________
☐ Altair 680b ☐ Kit ☐ Assembled ☐ Other (specify)
enclose $8 for postage and handling
☐ Please send free information package.
NAME________
ADDRESS________
CITY________ STATE & ZIP________

mits

2450 Alamo SE/Albuquerque, NM 87106, 505-243-7821

NOTE: Altair is a trademark of MITS, Inc.

Price, specifications subject to change. Please allow up to 60 days for delivery.